

**PROTOTYPE TELEMETRI ALAT PENDETEKSI DINI KEBAKARAN HUTAN  
MENGUNAKAN ATMEGA8 DENGAN ANTARMUKA KOMPUTER  
PROYEK AKHIR**



Diajukan kepada Fakultas Teknik Universitas Negeri Yogyakarta  
Untuk Memenuhi Sebagian Persyaratan Guna Memperoleh Gelar Ahli Madya



Disusun Oleh :

**ENGGAR PRAJANGGA**

**NIM. 10507131027**

**PROGRAM STUDI TEKNIK ELEKTRONIKA  
JURUSAN PENDIDIKAN TEKNIK ELEKTRONIKA  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA**

**2015**

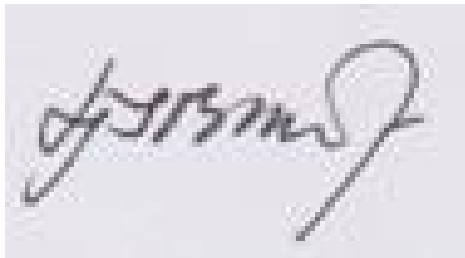
## LEMBAR PERSETUJUAN

Proyek akhir yang berjudul "*Prototype Telemetri Alat Pendeteksi Dini Kebakaran Hutan Menggunakan ATmega8 Dengan Antarmuka Komputer*" ini telah disetujui oleh pembimbing untuk diujikan.

Yogyakarta, 11 November 2015

Mengetahui

Koord. Prodi D3 Teknik Elektronika,



**Dr. Sri Waluyanti, M.Pd.**

**NIP. 19581218 198603 2 001**

Menyetujui

Dosen Pembimbing,



**Dessy Irmawati, S.T, M.T**

**NIP. 197912142010122 002**

## LEMBAR PENGESAHAN

Proyek akhir yang berjudul "Prototype Telemetri Alat Pendeteksi Dini Kebakaran Hutan Menggunakan ATmega8 Dengan Antarmuka Komputer" ini telah dipertahankan di depan dewan penguji pada tanggal 20 November 2015 dan dinyatakan lulus.

### DEWAN PENGUJI

Nama	Jabatan	Tanda Tangan	Tanggal
Dessy Irmawati, S.T, M.T	Ketua Penguji		07/01 2016
Satriyo Agung Dewanto, M.Pd.	Sekretaris Penguji		07/01 2016
Dr. Eko Marpanaji	Penguji Utama		07/01 2016

Yogyakarta, 9 Desember 2015

Fakultas Teknik Universitas Negeri Yogyakarta

Dekan,



Dr. Mochamad Bruri Triyono, M. Pd.

NIP. 19560216 198603 1 003

## LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan dibawah ini :

Nama : Enggar Prajangga

NIM : 10507131027

Program Studi : Teknik Elektronika

Judul PA : *Prototype* Telemetry Alat Pendeteksi Dini Kebakaran  
Hutan Menggunakan ATmega8 Dengan Antarmuka  
Komputer

Menyatakan bahwa Proyek Akhir ini adalah hasil pekerjaan saya sendiri dan sepanjang pengetahuan saya, tidak berisi materi yang ditulis orang lain sebagai persyaratan penyelesaian studi di Universitas Negeri Yogyakarta atau Perguruan Tinggi lain, kecuali bagian-bagian tertentu yang saya ambil sebagai acuan dengan mengikuti tata cara dan penulisan karya ilmiah yang lazim. Jika ternyata terbukti pernyataan ini tidak benar, sepenuhnya menjadi tanggung jawab saya.

Yogyakarta, 14 Oktober 2015

Penulis,



Enggar Prajangga

(NIM. 10507131027)

## **MOTTO**

*Hanya kebodohan meremehkan pendidikan*

*( P.Syrus )*

*Belajar dari kesalahan adalah kunci kesuksesan dimasa depan*

*( penulis )*

*Pengetahuan adalah kekuatan*

*( penulis )*

## **PERSEMBAHAN**

*Dengan penuh keyakinan karya ini Penulis persembahkan untuk:*

*Tuhan yang Maha Pengasih lagi Maha Penyayang.*

*Bapak, Ibu, kakak, adik dan seluruh keluarga besar atas doa dan dukungan yang selalu mendampingi.*

*Dosen Pembimbing Proyek Akhir, Ibu Dessy Irmawati, S.T, M.T yang selalu membimbing dan memotivasi untuk semangat dalam belajar dan penyelesaian proyek akhir ini.*

*Fitra Mega Kurniawan atas pengarahannya.*

*Rekan-rekan sahabat Kelas B 2010 PT Elektronika FT UNY.*

*Terimakasih atas dukungan, bantuan, inspirasi dan semangat kalian dalam penyelesaian Proyek Akhir ini.*

*Semua pihak yang telah membantu dalam proses pembuatan laporan ini.*

# **Prototype Telemetry Alat Pendeteksi Dini Kebakaran Hutan Menggunakan ATmega8 Dengan Antarmuka Komputer**

Oleh: Enggar Prajangga

Nim: 10507131027

## **ABSTRAK**

Pembuatan proyek akhir ini bertujuan untuk merealisasikan perangkat keras (*hardware*) dan perangkat lunak (*software*), serta mengetahui unjuk kerjanya, Alat Pendeteksi Dini kebakaran hutan menggunakan ATmega8 dengan antarmuka komputer dibuat dari beberapa blok rangkaian yang fungsional dengan tujuan agar mudah dalam perakitan dan pengoperasian saat digunakan, alat ini dikhususkan untuk membantu pos pemantau di sekitar hutan.

Metode perancangan alat pendeteksi dini kebakaran hutan dengan antarmuka komputer yang terdiri dari beberapa tahap yaitu, (1) Identifikasi kebutuhan, (2) Analisis kebutuhan, (3) Perancangan Alat, (4) Pembuatan alat, dan (5) Pengujian. Prinsip kerja alat pendeteksi dini kebakaran hutan menggunakan ATmega8 dengan antarmuka komputer yaitu terdiri dari blok-blok rangkaian fungsional yang dirakit menjadi satu kesatuan sehingga dapat digunakan sebagai pemantau keadaan hutan yang dipantau.

Hasil pengujian telah didapat bahwa perangkat keras yang terdiri dari *transmitter* dan *receiver*. Rangkaian *transmitter* terdiri dari sensor DHT11 yang dikontrol oleh sistem minimum ATmega8 yang menggunakan RF Modem KYL 200U untuk media pengirimnya dan mendapat *supply* dari adaptor 12V, sedangkan rangkaian *receiver* terdiri dari USB to TTL yang menggunakan RF Modem KYL 200U sebagai media penerimanya dan menggunakan komputer/laptop sebagai *interface*. Unjuk kerja alat ini adalah sebagai pemantau jarak jauh yang menggunakan komputer/laptop yang telah terpasang aplikasi *interface* sebagai pemantaunya yang bisa menyimpan data hasil pemantauan di Datalog.

Kata kunci : Telemetry, Kebakaran Hutan, RF Modem KYL 200U, DHT11, Mikrokontroler, Komputer/laptop

## KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa yang telah melimpahkan rahmat serta hidayah-Nya, sehingga penulis dapat menyelesaikan laporan proyek akhir ini dengan judul “*Prototype* Telemetry Alat Pendeteksi Dini Kebakaran Hutan Menggunakan ATmega8 Dengan Antarmuka Komputer”. Pembuatan Proyek Akhir ini sebagai syarat untuk memperoleh gelar Ahli Madya di Fakultas Teknik Universitas Negeri Yogyakarta.

Penulis menyadari sepenuhnya keberhasilan Proyek Akhir ini tidak lepas dari bantuan berbagai pihak, baik itu secara langsung maupun tidak langsung. Dengan kerendahan hati, pada kesempatan ini penulis mengucapkan rasa terimakasih yang sebesar-besarnya kepada :

1. Ibu Dessy Irmawati, S.T, M.T selaku Dosen Pembimbing Proyek Akhir yang telah membantu dan selalu memberi motivasi agar cepat lulus serta arahnya dalam bimbingan.
2. Bapak Muhammad Munir, M.Pd selaku Ketua Jurusan Pendidikan Teknik Elektronika Fakultas Teknik Universitas Negeri Yogyakarta.
3. Ibu Dr. Sri Waluyanti, M.Pd selaku Kaprodi Teknik Elektronika
4. Bapak Dr. Moch Bruri Triyono selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta
5. Orang Tua yang selalu mendoakan dan selalu memberikan dukungannya.
6. Para Dosen dan Staff Jurusan Pendidikan Teknik Elektronika yang telah memberikan bantuan sehingga terselesaikannya Proyek Akhir ini.
7. Seluruh teman-teman seangkatan dan seperjuangan, terima kasih atas kerja samanya serta dukungannya.
8. Semua pihak yang telah membantu Proyek Akhir ini.



Penulis menyadari dalam pembuatan Proyek Akhir ini masih jauh dari sempurna. Oleh karena itu saran dan kritik yang sangat membangun dibutuhkan guna menyempurnakan laporan Proyek Akhir ini. Semoga Proyek Akhir ini dapat memberikan manfaat bagi Siapa saja yang membacanya.

Yogyakarta, 14 Oktober 2015

A handwritten signature in dark ink, appearing to be 'Glen', with a horizontal line and a small mark at the end.

Penulis

## DAFTAR ISI

Halaman

HALAMAN JUDUL

**LEMBAR PERSETUJUAN ..... ii**

**LEMBAR PENGESAHAN ..... iii**

**LEMBAR PERNYATAAN KEASLIAN..... iv**

**MOTTO ..... v**

**PERSEMBAHAN..... vi**

**ABSTRAK ..... vii**

**KATA PENGANTAR ..... viii**

**DAFTAR ISI ..... x**

**DAFTAR GAMBAR..... xii**

**DAFTAR TABEL ..... xiv**

**DAFTAR LAMPIRAN ..... xv**

**BAB I PENDAHULUAN ..... 1**

A. Latar Belakang Masalah..... 1

B. Identifikasi Masalah..... 3

C. Batasan Masalah ..... 3

D. Rumusan Masalah ..... 4

E. Tujuan ..... 4

F. Manfaat..... 5

G. Keaslian Gagasan..... 5

**BAB II KAJIAN PUSTAKA ..... 8**

A. Pengertian Telemetri..... 8

B. Perangkat Keras (*Hardware*)..... 10

C.	Perangkat Lunak ( <i>Software</i> ).....	31
D.	Pengertian Suhu dan Kelembaban.....	44
<b>BAB III KONSEP RANCANGAN .....</b>		<b>48</b>
A.	Identifikasi Kebutuhan .....	48
B.	Analisis Kebutuhan .....	48
C.	Perancangan Alat.....	51
D.	Perancangan <i>Software</i> .....	59
E.	Alat dan Bahan.....	60
<b>BAB IV PENGUJIAN DAN PEMBAHASAN .....</b>		<b>68</b>
A.	Hasil Pengujian .....	68
B.	Pembahasan.....	71
C.	Unjuk Kerja .....	93
<b>BAB V KESIMPULAN DAN SARAN .....</b>		<b>95</b>
A.	Kesimpulan .....	95
B.	Keterbatasan Alat.....	96
C.	Saran .....	96
<b>DAFTAR PUSTAKA.....</b>		<b>97</b>
<b>LAMPIRAN.....</b>		<b>98</b>

## DAFTAR GAMBAR

	Halaman
Gambar 1. Blok Diagram Pada Telemetry .....	9
Gambar 2. ATmega8 .....	10
Gambar 3. Konfigurasi Pin ATmega8.....	11
Gambar 4. Blok Diagram ATmega8 .....	14
Gambar 5. Bit Global Interrupt Enable .....	15
Gambar 6. Peta Memori ATmega .....	17
Gambar 7. Sensor Kelembaban Udara/Humidity (DHT11).....	22
Gambar 8. Bentuk Fisik PCB .....	24
Gambar 9. KYL 200U .....	25
Gambar 10. Dimensi Instalasi Pada KYL 200U.....	27
Gambar 11. USB to TTL UART PL2303HXD .....	30
Gambar 12. Tampilan Microsoft Visual Studio 2010 .....	33
Gambar 13. Blok Diagram pada Board Arduino .....	36
Gambar 14. Board Arduino .....	36
Gambar 15. Interface dari Arduino IDE .....	37
Gambar 16. Diagram Blok Sederhana dari ATmega 328 .....	39
Gambar 17. Bagian-Bagian pada Papan Arduino Uno .....	40
Gambar 18. Komponen LED yang Terdapat pada Board Arduino Uno .....	44
Gambar 19. Blok Diagram Rangkaian.....	51
Gambar 20. Rangkaian Sistem Minimum ATmega8 .....	53
Gambar 21. Rangkaian Penurun Tegangan.....	54
Gambar 22. Rangkaian USB to TTL .....	54
Gambar 23. Flowchart Pada RX .....	56
Gambar 24. Flowchart Pada TX.....	58

Gambar 25. Interface saat Mengidentifikasi Port .....	70
Gambar 26. Interface pada Komputer.....	77
Gambar 27. Datalog pada Interface saat Merekam Data .....	78
Gambar 28. Pengujian pada Sistem Minimum ATmega8.....	81
Gambar 29. Pengujian saat Sensor DHT11 Bekerja .....	82
Gambar 30. Proses Loading Sensor selama 10 Detik.....	83

## DAFTAR TABEL

	Halaman
Tabel 1. <i>Port</i> C pada ATmega8 .....	12
Tabel 2. <i>Port</i> D pada ATmega8 .....	13
Tabel 3. Tabel Karakteristik Sensor Kelembaban Udara/ <i>Humidity</i> .....	23
Tabel 4. Spesifikasi Teknis pada KYL 200U .....	26
Tabel 5. Definisi Pin pada KYL 200U .....	28
Tabel 6. Tabel Karakteristik Rangkaian pada <i>Board</i> Arduino Uno.....	41
Tabel 7. Jadwal Pelaksanaan .....	62
Tabel 8. Komponen dan Biaya Proyek Akhir.....	63
Tabel 9. Pengujian Kerja Suplai Tegangan.....	64
Tabel 10. Pengujian Fungsional dari Alat.....	65
Tabel 11. Pengukuran pada IC Regulator .....	68
Tabel 12. Pengukuran pada Sensor dan Transmitter.....	71
Tabel 13. Pengujian Alat.....	73
Tabel 14. Jarak Koneksi KYL 200U Kondisi LOS.....	79
Tabel 15. Jarak Koneksi KYL 200U Kondisi NLOS .....	80

## DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Rangkaian Keseluruhan .....	98
Lampiran 2. <i>Layout</i> Komponen .....	99
Lampiran 3. <i>Layout</i> PCB.....	100
Lampiran 4. PROGRAM .....	101
Lampiran 5. <i>Datasheet</i> ATmega8 .....	107
Lampiran 6. <i>Datasheet</i> DHT11 .....	123
Lampiran 7. <i>Datasheet</i> RF Modem KYL 200U .....	131
Lampiran 8. <i>Datasheet</i> USB to TTL Converter.....	146

# **BAB I**

## **PENDAHULUAN**

### **A. LATAR BELAKANG MASALAH**

Hutan merupakan suatu pondasi alam dalam menyediakan dan mengendalikan berbagai kebutuhan manusia, seperti udara, air dan sebagainya. Namun, bersamaan itu pula sebagai dampak negatif atas pengelolaan hutan yang eksploitatif dan tidak berpihak pada kepentingan rakyat, pada akhirnya menyisakan banyak persoalan, diantaranya tingkat kerusakan hutan yang mengkhawatirkan. Demikian juga halnya di Indonesia, permasalahan perusakan hutan yang akibatnya tidak saja dirasakan oleh masyarakat sekitar, tetapi juga meliputi aspek lepas batas negara, sehingga merugikan masyarakat negara lain. Kebakaran hutan di Indonesia memberikan akibat terjadinya pencemaran udara di beberapa negara di kawasan ASEAN, disebabkan kebakaran hutan tidak hanya melingkupi satu negara tetapi sudah meluas ke negara ASEAN lainnya, maka pelaksanaan pengendalian hutan tersebut dilakukan melalui bentuk kerjasama sesama anggota ASEAN. Dampak langsung dari kebakaran hutan tersebut antara lain:

1. Timbulnya infeksi saluran pernapasan akut bagi masyarakat.
2. Berkurangnya efisiensi kerja karena saat terjadi kebakaran hutan dalam skala besar, sekolah-sekolah dan kantor-kantor akan diliburkan.
3. Terganggunya transportasi di darat, laut maupun udara.
4. Timbulnya persoalan internasional asap dari kebakaran hutan tersebut yang menyebabkan kerugian materiil dan imateriil pada masyarakat setempat dan sering kali menimbulkan pencemaran asap lintas batas



(*transboundary haze pollution*) ke wilayah negara-negara tetangga, seperti Malaysia dan Singapura. Asap dari kebakaran hutan dan lahan itu ternyata telah menurunkan kualitas udara dan jarak pandang di region Sumatera dan Kalimantan, termasuk Malaysia, Singapura, Brunei dan sebagian Thailand (Suratmo, 2003).

Melihat dari latar belakang tersebut, pada pembuatan tugas akhir ini alat yang akan dibuat ini akan menggunakan RF Modem KYL 200U *wireless* sebagai pengirim nirkabelnya, sensor DHT11 (*Digital Humidity and Temperature sensor*) sebagai sensornya, ATmega8 dipilih sebagai kontrolnya, dan menggunakan komputer/*notebook* untuk memantaunya. Pada proyek akhir ini dipilih ATmega8 karena harganya yang murah dan kapasitas *memory* 10kb yang tersedia sudah mencukupi untuk memuat program. Permasalahan yang muncul ini diantaranya adalah bagaimana merancang sistem minimum untuk menempatkan mikrokontroler ATmega8 sebagai otak untuk mengaplikasikan RF Modem KYL 200U *wireless* dalam proses mengirim dan menerima data, sedangkan pada sensor DHT11 yang digunakan untuk mengukur suhu dan kelembaban yang ditempatkan pada *transceiver* dari RF Modem KYL 200U *wireless* sedangkan pada *receiver* yang digunakan ke komputer yaitu dengan menambahkan USB (*Universal Serial Bus*) konektor yang akan dirancang menggunakan USB to TTL (*Transistor-Transistor Logic*), untuk tahap akhir dibuat *software* yang akan merekam data perubahan suhu dan kelembaban per detik yang akan tercatat secara kontinu di *notepad*.

## B. IDENTIFIKASI MASALAH

Berdasarkan uraian latar belakang masalah diatas, maka dapat dibuat suatu identifikasi masalah sebagai berikut :

1. Belum adanya *prototype* telemetri alat pendeteksi dini kebakaran hutan yang menggunakan sensor DHT11 yang bisa mendeteksi suhu dan kelembaban.
2. Belum adanya *prototype* telemetri alat pendeteksi dini kebakaran hutan yang memanfaatkan RF Modem KYL 200U sebagai pemberi informasi kebakaran hutan ke pengguna dari jarak jauh.
3. Belum adanya *prototype* telemetri alat pendeteksi dini kebakaran hutan yang dapat memberikan hasil data nyata dengan *data logger* dan menampilkannya di *notepad* pada komputer.

## C. BATASAN MASALAH

Berdasarkan indentifikasi masalah yang didapat perlunya adanya pembatasan masalah, diantaranya:

1. Mikrokontroller yang digunakan adalah ATmega8.
2. Komunikasi serial antara mikrokontroller dengan komputer atau sebaliknya dengan menggunakan *protocol* USB to TTL.
3. Alat yang dibuat adalah *prototype* telemetri alat pendeteksi dini kebakaran hutan dengan sensor DHT 11 menggunakan RF Modem KYL 200U.
4. Perangkat antarmuka yang digunakan adalah PC (*Personal Computer*) dan komputer/laptop.

#### D. RUMUSAN MASALAH

Berdasarkan latar belakang masalah, identifikasi masalah, dan batasan masalah, maka di dapat rumusan masalah sebagai berikut :

1. Bagaimana merancang *hardware* dari *prototype* telemetri alat pendeteksi dini kebakaran hutan menggunakan RF Modem KYL 200U sebagai *transceiver* yang ada pada RF Modem KYL 200U, yang akan mengirimkan data cuaca yang terbaca dalam komputer.
2. Bagaimana merancang *software* dari *prototype* telemetri alat pendeteksi dini kebakaran hutan dengan membuat aplikasi menggunakan Visual Studio 2010 (Visual Basic) untuk mendapatkan hasil data yang dimonitoring.
3. Bagaimana unjuk kerja dari *prototype* telemetri alat pendeteksi dini kebakaran hutan dengan memanfaatkan RF Modem KYL 200U sebagai *transceiver* yang kemudian dipantau menggunakan komputer.

#### E. TUJUAN

1. Mengetahui rancangan *hardware* dari *prototype* telemetri alat pendeteksi dini kebakaran hutan yang menggunakan sensor DHT11 untuk mendeteksi perubahan suhu dan kelembaban.
2. Mengetahui rancangan *hardware* dari *prototype* telemetri alat pendeteksi dini kebakaran hutan yang menggunakan RF Modem KYL 200U *wireless* sebagai media pengirim dan penerima, yang akan mengirimkan data cuaca yang terbaca dalam komputer.
3. Mengetahui unjuk kerja dari *prototype* telemetri alat pendeteksi dini kebakaran hutan yang memanfaatkan RF Modem KYL 200U *wireless*

sebagai media pengirim dan penerima yang kemudian dipantau menggunakan komputer.

#### **F. MANFAAT**

Dalam pembuatan alat ini, di harapkan dapat memberikan manfaat sebagai berikut :

1. Bagi mahasiswa
  - a. Sebagai sarana implementasi pengetahuan yang di dapat di bangku pendidikan.
  - b. Sebagai bentuk kontribusi terhadap Universitas baik dalam citra maupun daya tawar terhadap masyarakat luas.
2. Bagi Jurusan Pendidikan Teknik Elektronika
  - a. Terciptanya alat yang inovatif dan bermanfaat sebagai sarana ilmu pengetahuan.
  - b. Sebagai wujud partisipasi dalam pengembangan dibidang IPTEK.
3. Bagi Dunia Usaha dan Dunia Industri
  - a. Terciptanya alat sebagai sarana peningkatan teknologi dalam dunia industri.
  - b. Sebagai bentuk kontribusi terhadap industri dalam mewujudkan pengembangan teknologi.

#### **G. KEASLIAN GAGASAN**

Pada penelitian tahun 2008 telah dibuat sebuah proyek akhir alat pendeteksi dini kebakaran hutan dengan menggunakan modem wavecom sebagai pengirim sms berbasis mikrokontroler ATmega16. Alat ini berfungsi untuk mengirim informasi terhadap petugas penjaga hutan dan juga dapat membantu pihak terkait untuk menanggulangi dan mencegah meluasnya

kebakaran hutan secara cepat. Saat terjadinya kebakaran hutan, maka data yang terbaca akan diteruskan ke modem wavecom langsung mengirimkan sms (Setiawan, Muhammad. 2008).

Pada proyek akhir tahun 2009 dibuat sebuah sistem informasi tentang kebakaran hutan yang berbasis Web SIG. Sistem ini memberikan informasi tentang sistem peringatan dini, pencegahan, pemadaman dan penanganan setelah kebakaran hutan terjadi. Data satelit dan dokumen-dokumen yang diekstraksi dari berbagai sumber, kemudian diolah dengan pendekatan Sistem Informasi Geografis (SIG) sehingga didapatkan informasi dan solusi pencegahan, penanganan pada saat terjadi dan pasca kebakaran hutan. Menggunakan SIG akan lebih mudah bagi para pengambil keputusan untuk menganalisa data tentang kebakaran hutan meliputi data hotspot, data emergency dan evakuasi serta data kecocokan vegetasi dengan lahan yang sudah terbakar. Karena sistem informasi ini berbasis web, maka digunakan teknologi yang sesuai yaitu Map Server sebagai web server, php, html, dan javascript sebagai pembangun sistem dan basis data Postgre SQL sebagai penyimpanan data. Semua informasi yang dihasilkan dalam sistem ini akan divisualisasikan dalam peta-peta dan informasi mengenai kebakaran hutan (Wahyu, Catur. 2009)

Pada penelitian tahun 2012, dikembangkan sistem pendeteksi dini kebakaran hutan. Sistem ini dirancang dengan penanggulangan khusus dan pengiriman informasi melalui pesan pendek (SMS) dan panggilan berbasis mikrokontroler, untuk menciptakan sistem keamanan khususnya bahaya kebakaran pada bangunan (gedung, pertokoan maupun rumah) yang sangat penting. Perangkat keras yang digunakan dalam sistem ini terdiri dari

mikrokontroler Atmega8 sebagai pusat kendali, rangkaian sensor MQ-2 sebagai pendeteksi asap, dan rangkaian flame sensor sebagai deteksi api, rangkaian TTL Rs 232, rangkaian relay dan sirine, serta modem wavecom sebagai pengirim SMS dan telepon ke pengguna (*user*). Ketika sensor MQ-2 mendeteksi adanya asap, maka modem wavecom akan mengirimkan SMS ke pemilik rumah, dan jika flame sensor mendeteksi adanya api, maka modem wavecom akan melakukan panggilan telepon (Agus, 2012).

Melihat kekurangan dari alat yang telah dibuat tersebut, maka penulis akan sedikit memodifikasi alat dengan mengganti media sms maupun yang berbasis Web SIG dengan sistem *transceiver* menggunakan modul KYL 200U. Dimana kekurangan saat menggunakan sms maupun Web SIG memungkinkan terjadi gangguan yang mengakibatkan penerimaan sinyal terganggu dan sms terlambat, hal ini dapat diatasi dengan sistem *transceiver* karena data akan dikirimkan langsung tanpa perantara sinyal dari satelit. Proyek akhir ini berjudul “ **PROTOTYPE TELEMETRI ALAT PENDETEKSI DINI KEBAKARAN HUTAN MENGGUNAKAN ATMEGA8 DENGAN ANTARMUKA KOMPUTER**” yang merupakan asli gagasan penulis. Adapun yang menjadi ciri khas pada proyek akhir ini adalah :

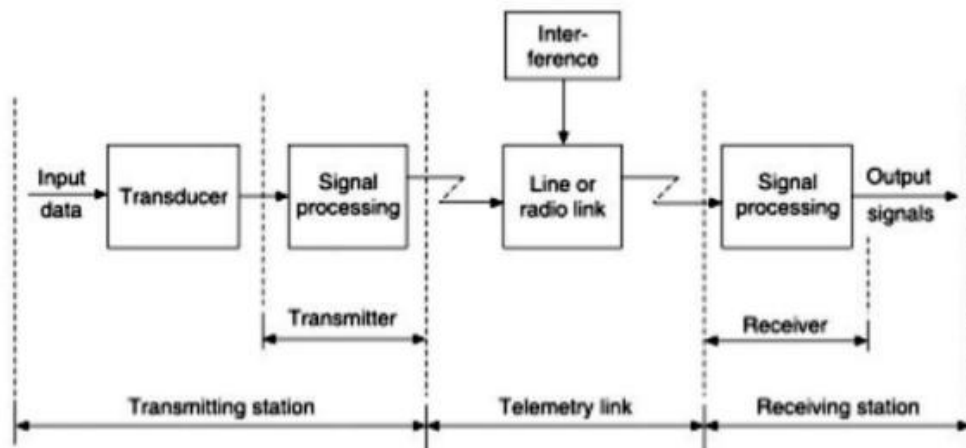
- a. Menggunakan *box* plastik sebagai tempat dari alatnya.
- b. Menggunakan RF Modem KYL 200U *wireless* sebagai media pengirim dan penerima nirkabelnya.
- c. Menggunakan ATmega8 sebagai sistem pengontrolnya.
- d. Menggunakan DHT11 sebagai sensor suhu dan kelembabannya.
- e. Menggunakan PC (*Personal Computer*) atau *notebook* sebagai sarana antarmukanya.

## **BAB II**

### **KAJIAN PUSTAKA**

#### **A. PENGERTIAN TELEMETRI**

Telemetry adalah sebuah teknologi pengukuran yang dilakukan dari jarak jauh dan melaporkan informasi kepada perancang atau operator sistem. Kata telemetry berasal dari bahasa Yunani yaitu *tele* artinya jarak jauh sedangkan *metron* artinya pengukuran. Secara istilah telemetry diartikan sebagai suatu bidang keteknikan yang memanfaatkan instrumen untuk mengukur panas, radiasi, kecepatan atau properti lainnya dan mengirimkan data hasil pengukuran ke penerima yang letaknya jauh secara fisik, berada diluar dari jangkauan pengamat atau *user*. Perbedaan telekomunikasi dengan telemetry adalah jika telekomunikasi itu adalah hubungan dalam cakupan luas sedangkan telemetry itu merupakan salah satu sub telekomunikasi yang sebagai hubungan dalam bentuk pengukuran. Telemetry dalam keadaan bergerak berpengaruh pada saat pengukuran, pengukuran tersebut untuk mendapatkan nilai percepatan pada suatu benda bergerak. Telemetry bergerak sangat rentan terhadap *noise*. *Noise* yang sering terjadi adalah *noise* dari getaran, suhu, tekanan atmosfer, dan benda yang menjadi penghalang (P.H. Simale, 1995).



Gambar 1. Blok Diagram Pada Telemetri

Pada gambar 1 menunjukkan blok diagram telemetri. Blok diagram tersebut pada setelah pengambilan data akan masuk ke *transducer*. *Transducer* merupakan komponen yang bertugas merubah data menjadi bentuk informasi dan menempatkannya pada media transmisi. Setelah melalui *transducer* akan masuk ke *signal processing*. *Signal processing* ini berfungsi untuk mengubah informasi yang akan dikirim/diterima menjadi bentuk yang sesuai dengan media transmisi yang digunakan. Setelah itu akan masuk ke proses telemetri atau proses pengiriman jarak jauh yang akan diterima oleh penerima yang juga melalui *signal processing* dan selanjutnya akan dikeluarkan data yang diterima dari pengirim.



## B. PERANGKAT KERAS (*HARDWARE*)

### 1. Mikrokontroller ATmega8



Gambar 2. ATmega8

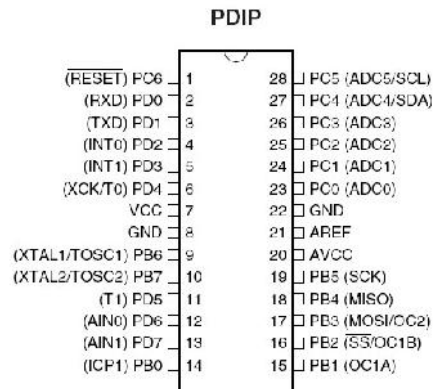
(Sumber: [black-electronics.com](http://black-electronics.com))

Gambar 2 adalah ATmega8 yang merupakan salah satu jenis mikrokontroller yang di dalamnya terdapat berbagai macam fungsi. Perbedaannya pada mikro yang pada umumnya digunakan seperti MCS51 adalah pada ATmega8 tidak perlu menggunakan *oscillator eksternal* karena di dalamnya sudah terdapat *internal oscillator*. Selain itu kelebihan dari ATmega8 adalah memiliki *Power-On Reset*, yaitu tidak perlu ada tombol *reset* dari luar karena cukup hanya dengan mematikan *supply*, maka secara otomatis ATmega8 akan melakukan *reset*. Untuk beberapa jenis ATmega8 terdapat beberapa fungsi khusus seperti ADC, EEPROM sekitar 128 *byte* sampai dengan 512 *byte*.

ATmega8 adalah mikrokontroller CMOS 8-bit berarsitektur AVRISC yang memiliki 8K *byte in-System Programmable Flash*. Mikrokontroller dengan konsumsi daya rendah ini mampu mengeksekusi instruksi dengan kecepatan maksimum 16MIPS pada frekuensi 16MHz. Jika dibandingkan dengan Atmega8L perbedaannya hanya terletak pada besarnya tegangan yang diperlukan untuk bekerja. Pada ATmega8 tipe L, mikrokontroller ini

dapat bekerja dengan tegangan antara 2,7-5,5 V sedangkan untuk ATmega8 hanya dapat bekerja pada tegangan antara 4,5-5,5 V (Winoto, Ardi. 2008).

a. Konfigurasi Pin ATmega8



Gambar 3. Konfigurasi *Pin* ATmega8

(Sumber: [www.raharja.ac.id](http://www.raharja.ac.id))

Gambar 4 menjelaskan konfigurasi dari setiap *pin* dari ATmega8 yang memiliki 28 *Pin*, masing-masing *pin* memiliki fungsi yang berbeda-beda baik sebagai *port* maupun fungsi yang lainnya. Berikut akan dijelaskan fungsi dari masing-masing kaki ATmega8 :

1) VCC

Merupakan *supply* tegangan digital.

2) GND

Merupakan *ground* untuk semua komponen yang membutuhkan *grounding*.

### 3) Port C (PC5...PC0)

Tabel 1. Port C pada ATmega8

Port Pin	Alternate Function
PC6	RESET (Reset pin)
PC5	ADC5 (ADC Input Channel 5) SCL (Two-wire Serial Bus Clock Line)
PC4	ADC4 (ADC Input Channel 4) SDA (Two-wire Serial Bus Data Input/Output Line)
PC3	ADC3 (ADC Input Channel 3)
PC2	ADC2 (ADC Input Channel 2)
PC1	ADC1 (ADC Input Channel 1)
PC0	ADC0 (ADC Input Channel 0)

Tabel 1 menjelaskan Port C yang merupakan sebuah 7-bit *bi-directional I/O port* yang di dalam masing-masing *pin* terdapat *pull-up* resistor. Jumlah *pin* nya hanya 7 buah mulai dari *pin* C.0 sampai dengan *pin* C.6. Sebagai keluaran *output port* C memiliki karakteristik yang sama dalam hal menyerap arus (*sink*) ataupun mengeluarkan arus (*source*).

### 4) RESET/PC6

Jika RSTDISBL Fuse diprogram, maka PC6 akan berfungsi sebagai *pin I/O*. *Pin* ini memiliki karakteristik yang berbeda dengan setiap *pin* yang terdapat pada port C lainnya. Namun jika RSTDISBL Fuse tidak diprogram, maka *pin* ini akan berfungsi sebagai *input reset*. Dan jika *level* tegangan yang masuk ke *pin* ini rendah dan pulsa yang ada lebih pendek dari pulsa minimum, maka akan menghasilkan suatu kondisi *reset* meskipun *clock*-nya tidak bekerja.

### 5) Port D (PD7...PD0)

Tabel 2 menjelaskan Port D yang merupakan 8-bit *bi-directional I/O* dengan internal *pull-up* resistor. Fungsi dari port ini

sama dengan setiap *port* yang lain. Hanya saja pada *port* ini tidak terdapat kegunaan-kegunaan yang lain. Pada *port* ini hanya berfungsi sebagai masukan dan keluaran saja atau biasa disebut dengan I/O.

Tabel 2. *Port D* pada ATmega8

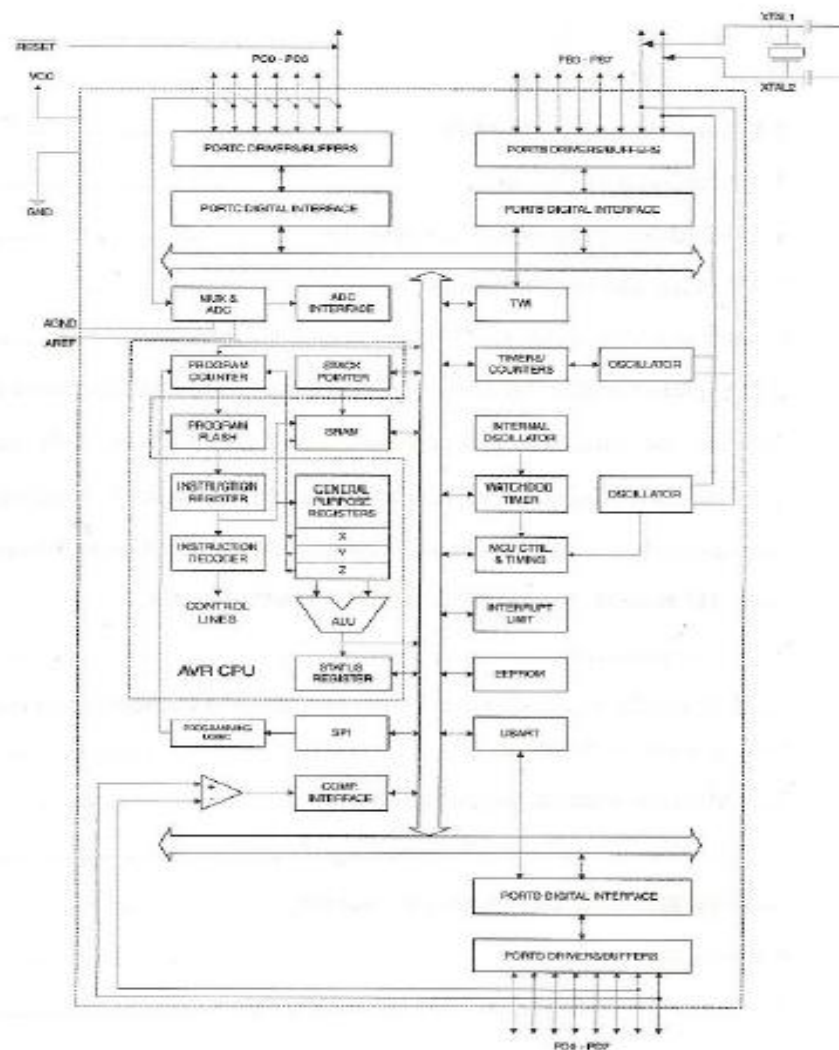
Port Pin	Alternate Function
PD7	AIN1 (Analog Comparator Negative Input)
PD6	AIN0 (Analog Comparator Positive Input)
PD5	T1 (Timer/Counter 1 External Counter Input)
PD4	XCK (USART External Clock Input/Output) T0 (Timer/Counter 0 External Counter Input)
PD3	INT1 (External Interrupt 1 Input)
PD2	INT0 (External Interrupt 0 Input)
PD1	TXD (USART Output Pin)
PD0	RXD (USART Input Pin)

#### 6) AVcc

*Pin* ini berfungsi sebagai *supply* tegangan untuk ADC. Untuk *pin* ini harus dihubungkan secara terpisah dengan VCC karena *pin* ini digunakan untuk analog saja. Bahkan jika ADC pada AVR tidak digunakan tetap saja disarankan untuk menghubungkannya secara terpisah dengan VCC. Jika ADC digunakan, maka AVcc harus dihubungkan ke VCC melalui *low pass filter*.

#### 7) AREF

Merupakan *pin* referensi jika menggunakan ADC.



Gambar 4. Blok Diagram ATmega8

(Sumber: [www.circuitstoday.com](http://www.circuitstoday.com))

Pada gambar 5 menjelaskan blok diagram dari ATmega8. Dibagian AVR status *register* mengandung beberapa informasi mengenai hasil dari kebanyakan hasil eksekusi instruksi aritmatik. Informasi ini digunakan untuk altering arus program sebagai kegunaan untuk meningkatkan performa pengoperasian. *Register* ini di-*update* setelah operasi ALU (*Arithmetic Logic Unit*) hal

tersebut seperti yang tertulis dalam *datasheet* khususnya pada bagian *Instruction Set Reference*. Dalam hal ini untuk beberapa kasus dapat membuang penggunaan kebutuhan instruksi perbandingan yang telah didedikasikan serta dapat menghasilkan peningkatan dalam hal kecepatan dan kode yang lebih sederhana dan singkat. *Register* ini tidak secara otomatis tersimpan ketika memasuki sebuah rutin interupsi dan juga ketika menjalankan sebuah perintah setelah kembali dari interupsi. Namun hal tersebut harus dilakukan melalui *software*.

#### 8) Bit 7(I)

Gambar 6 menjelaskan tentang *bit Global Interrupt Enable*. *Bit* ini harus di-*set* agar semua perintah interupsi dapat dijalankan. Untuk perintah interupsi individual akan di jelaskan pada bagian yang lain.

Bit	7	6	5	4	3	2	1	0	
	I	T	H	S	V	N	Z	C	SREG
Read/write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Gambar 5. *Bit Global Interrupt Enable*

(Sumber: [www.raharja.ac.id](http://www.raharja.ac.id))

Jika *bit* ini di-*reset*, maka semua perintah interupsi baik yang individual maupun yang secara umum akan di abaikan. *Bit* ini akan dibersihkan atau *cleared* oleh *hardware* setelah sebuah instruksi di jalankan dan akan di-*set* kembali oleh perintah RETI.

*Bit* ini juga dapat di-set dan di-reset melalui aplikasi dan instruksi SEI dan CLL.

9) *Bit 6(T)*

Merupakan *bit Copy Storage*. Instruksi *bit Copy Instructions* BLD (*Bit Load*) and BST (*Bit Store*) menggunakan *bit* ini sebagai asal atau tujuan untuk *bit* yang telah dioperasikan. Sebuah *bit* dari sebuah *register* dalam *Register File* dapat disalin ke dalam *bit* ini dengan menggunakan instruksi BST, dan sebuah *bit* di dalam *bit* ini dapat disalin ke dalam *bit* di dalam *register* pada *Register File* dengan menggunakan perintah BLD.

10) *Bit 5(H)*

Merupakan *bit Half Carry Flag*. *Bit* ini menandakan sebuah *Half Carry* dalam beberapa operasi aritmatika. *Bit* ini berfungsi dalam aritmatika BCD.

11) *Bit 4(S)*

Merupakan *Sign bit*. *Bit* ini selalu merupakan sebuah eksklusif di antara *Negative Flag (N)* dan *two's Complement Overflow Flag (V)*.

12) *Bit 3(V)*

Merupakan *bit Two's Complement Overflow Flag*. *Bit* ini menyediakan fungsi aritmatika dua komplemen.

13) *Bit 2(N)*

Merupakan *bit Negative Flag*. *Bit* ini mengindikasikan sebuah hasil *negative* di dalam sebuah fungsi logika atau aritmatika.

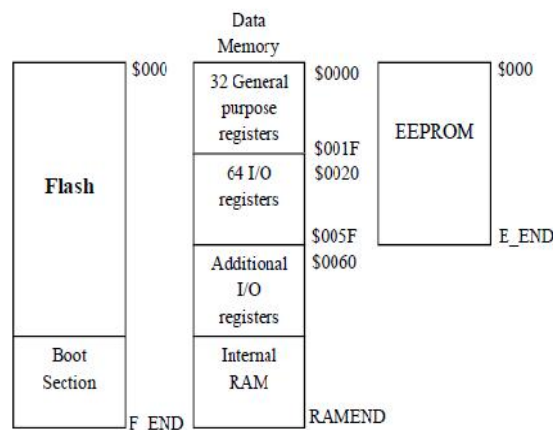
14) *Bit 1(Z)*

Merupakan *bit Zero Flag*. *Bit* ini mengindikasikan sebuah hasil nol “0” dalam sebuah fungsi aritmatika atau logika.

15) 0(C)

Merupakan *bit Carry Flag*. *Bit* ini mengindikasikan sebuah *Carry* atau sisa dalam sebuah aritmatika atau logika.

## b. Memori AVR ATmega



Gambar 6. Peta Memori ATmega

Gambar 6 menjelaskan peta memori ATmega yang terbagi menjadi tiga yaitu :

### 1) Memori *Flash*

Memori *flash* adalah memori ROM tempat kode-kode program berada. Kata *flash* menunjukkan jenis ROM yang dapat ditulis dan dihapus secara elektrik. Memori *flash* terbagi menjadi dua bagian, yaitu bagian aplikasi dan bagian *boot*. Bagian aplikasi adalah bagian kode-kode program aplikasi berada. Bagian *boot* adalah bagian yang digunakan khusus untuk *booting* awal yang dapat diprogram untuk menulis bagian aplikasi tanpa melalui



*programmer/downloader*, misalnya melalui USART (Winoto, Ardi. 2008).

## 2) Memori Data

Memori data adalah memori RAM yang digunakan untuk keperluan program. Memori data terbagi menjadi empat bagian, yaitu 32 GPR (*General Purpose Register*) adalah *register* khusus yang bertugas untuk membantu eksekusi program oleh ALU (*Arithmetic Logic Unit*), dalam instruksi *assembler* setiap instruksi harus melibatkan GPR. Dalam bahasa C biasanya digunakan untuk variabel global atau nilai balik fungsi dan nilai-nilai yang dapat memperingan kerja ALU. Dalam istilah *processor* komputer sehari-hari GPR dikenal sebagai "*chace memory*". I/O register dan *additional I/O register* adalah *register* yang difungsikan khusus untuk mengendalikan berbagai *peripheral* dalam mikrokontroler seperti *pin port*, *timer/counter*, *usart* dan lain-lain. *Register* ini dalam keluarga mikrokontroler MCS51 dikenal sebagai SFR (*Special Function Register*) (Winoto, Ardi. 2008).

## 3) EEPROM

EEPROM adalah memori data yang dapat mengendap ketika chip mati (*off*), digunakan untuk keperluan penyimpanan data yang tahan terhadap gangguan catu daya.

### c. Timer/Counter 0

*Timer/counter* 0 adalah sebuah *timer/counter* yang dapat mencacah sumber pulsa/clock baik dari dalam *chip* (*timer*)

ataupun dari luar *chip* (*counter*) dengan kapasitas 8-bit atau 256 cacahan. *Timer/counter* dapat digunakan untuk :

- 1) *Timer/counter* biasa
- 2) *Clear Timer on Compare Match* (selain ATmega8)
- 3) *Generator* frekuensi (selain ATmega8)
- 4) *Counter* pulsa *eksternal*

d. Komunikasi Serial Pada ATmega8

Mikrokontroler AVR ATmega8 memiliki *port* USART pada *pin* 2 dan *pin* 3 untuk melakukan komunikasi data antara mikrokontroler dengan mikrokontroler ataupun mikrokontroler dengan komputer. USART dapat difungsikan sebagai transmisi data sinkron, dan asinkron. Sinkron berarti *clock* yang digunakan antara *transmitter* dan *receiver* satu sumber *clock*. Sedangkan asinkron berarti *transmitter* dan *receiver* mempunyai sumber *clock* sendiri-sendiri. USART terdiri dalam tiga blok yaitu *clock generator*, *transmitter*, dan *receiver*.

## 2. Sensor

Sensor merupakan jenis transduser yang digunakan untuk mengubah besaran mekanis, magnetis, panas, sinar, dan kimia menjadi tegangan dan arus listrik. Sensor sering digunakan untuk pendeteksian pada saat melakukan pengukuran atau pengendalian. Beberapa jenis sensor yang banyak digunakan dalam rangkaian elektronik antara lain sensor cahaya (LDR), sensor suhu (LM35), dan sensor kelembaban udara (DHT11) (Janata,J.2009).

Di dalam melakukan analisa ini digunakan sensor tipe LM35 yang digunakan sebagai alat sensor suhu didalam rangkaian dengan menggunakan suplai tegangan sebesar 0-5V dari arduino yang merupakan *board* elektronik atau papan rangkaian elektronik *open source* yang di dalamnya terdapat komponen utama yaitu sebuah chip mikrokontroler dengan jenis AVR dari perusahaan Atmel. Mikrokontroler itu sendiri adalah *chip* atau IC (*Integrated Circuit*) yang bisa diprogram menggunakan komputer. Tujuan menanamkan program pada mikrokontroler adalah agar rangkaian elektronik dapat membaca *input*, memproses *input* tersebut dan kemudian menghasilkan *output* sesuai yang diinginkan.

Jadi mikrokontroler bertugas sebagai “otak” yang mengendalikan *input*, proses dan *output* sebuah rangkaian elektronik. Mikrokontroler ada pada perangkat elektronik di sekeliling kita, misalnya handphone, MP3 player, DVD, televisi, AC, dll. Mikrokontroler juga dipakai untuk keperluan mengendalikan robot, baik robot mainan, maupun robot industri. Karena komponen utama arduino adalah mikrokontroler, maka arduino pun dapat diprogram menggunakan komputer sesuai kebutuhan kita.

Pada saat mengukur suhu ruangan dibutuhkan satu komponen saja yaitu LM35. LM35 merupakan komponen elektronis berupa IC yang mampu menghasilkan tegangan sebanding dengan besaran suhu yang menyelimuti. Keluaran tegangan telah terkalibrasi dalam skala Kelvin. Linieritas keluaran tegangan pada skala Celsius adalah 10mV/°C dengan akurasi  $\pm 1/4^\circ\text{C}$  pada suhu ruang, dengan rentang suhu yang dapat direspon antara -55 sampai 150°C, penggunaan LM35 tidak

memerlukan perangkat tambahan, dengan mempunyai *output* impedansi yang rendah sehingga akan mempermudah dalam pembacaan dan kontrol. Konsumsi energi yang diperlukan IC ini sangat rendah 60 pA, sehingga tidak menimbulkan panas yang relatif besar atau kurang dari 0,1°C. Sensor ini bekerja pada jangka tegangan yang lebar (antara 4 sampai 30 Volt) yang memudahkan penyesuaian perancangan dengan perangkat sesudahnya yaitu *output* dapat langsung dihubungkan dengan *port* mikrokontroler yang memiliki ADC atau dengan arduino, yang didalamnya terdapat *port* ADC (*analog input*) sebanyak 6 buah dengan jenis, ADC yg dipakai adalah jenis ADC sebesar 10 *bit*. Mikrokontroler ada pada perangkat elektronik di sekeliling kita, misalnya handphone, MP3 player, DVD, televisi, AC, dll.

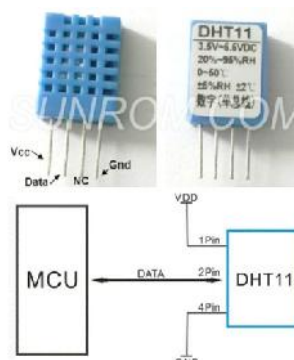
Dalam mendeteksi keadaan alam sekitar berupa data hasil analog *input* yang dibaca oleh sensor maka dibawah ini terdapat beberapa dasar teori yang menjadi acuan untuk mengenal cara kerja sensor yang digunakan dalam penelitian ini adalah sebagai berikut :

a. Sensor kelembaban udara/Humidity (DHT11)

Kelembaban udara menggambarkan kandungan uap air di udara yang dapat dinyatakan sebagai kelembaban mutlak, kelembaban nisbi (relatif) maupun defisit tekanan uap air. Kelembaban nisbi adalah membandingkan antara kandungan/tekanan uap air aktual dengan keadaan jenuhnya atau pada kapasitas udara untuk menampung uap air.

Peralatan elektronik juga menjadi mudah berkarat jika udara disekitarnya memiliki kelembaban yang cukup tinggi. Oleh karena itu,

informasi mengenai kelembaban udara pada suatu area tertentu menjadi sesuatu hal yang penting untuk diketahui karena menyangkut efek-efek yang ditimbulkannya. Informasi mengenai nilai kelembaban udara diperoleh dari proses pengukuran. Alat yang biasanya digunakan untuk mengukur kelembaban udara adalah higrometer.



Gambar 7. Sensor kelembaban udara/Humidity (DHT11)

(Sumber: [sunrom.com](http://sunrom.com))

Gambar 7 adalah sensor DHT11. Sensor DHT11 adalah sensor digital yang dapat mengukur suhu dan kelembaban udara di sekitarnya. Sensor ini sangat mudah digunakan bersama dengan arduino. Memiliki tingkat stabilitas yang sangat baik serta fitur kalibrasi yang sangat akurat. Koefisien kalibrasi disimpan dalam OTP program memori, sehingga ketika internal sensor mendeteksi sesuatu, maka modul ini menyertakan koefisien tersebut dalam kalkulasinya, DHT11 ini termasuk sensor yang memiliki kualitas terbaik, dinilai dari respon, pembacaan data yang cepat, dan kemampuan anti-*interference*. Ukurannya yang kecil, dan dengan transmisi sinyal hingga 20 meter, dengan spesifikasi : *Supply Voltage* : +5 V, *Temperature range* : 0-50 °C *error of*  $\pm 2$  °C, *Humidity* : 20-90% RH  $\pm 5\%$  RH *error*, dengan spesifikasi *digital interfacing system* yang demikian, maka akan

membuat produk ini cocok digunakan untuk banyak aplikasi-aplikasi pengukuran suhu dan kelembaban.

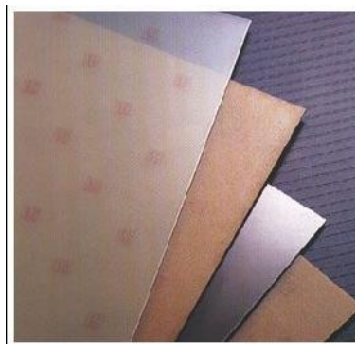
Tabel 3. Tabel karakteristik sensor kelembaban udara/*Humidity*

<b>Model</b>	DHT11
<b>Powersupply</b>	3-5.5V DC
<b>Output signal</b>	<i>Digital signal via single-bus</i>
<b>Measuring range</b>	<i>humidity 20-90% RH <math>\pm</math> 5% RH error</i> <i>temperature 0-50 °C error of <math>\pm</math> 2 °C</i>
<b>Accuracy</b>	<i>humidity <math>\pm</math> 4% RH (Max <math>\pm</math> 5% RH);</i> <i>temperature <math>\pm</math> 2° Celsius</i>
<b>Resolution or Sensitivity</b>	<i>humidity 1% RH; temperature 0.1 Celsius</i>
<b>Repeatability</b>	<i>humidity <math>\pm</math> 1% RH; temperature <math>\pm</math> 1° Celsius</i>
<b>Humidity hysteresis</b>	$\pm$ 1% RH
<b>Long-term Stability</b>	$\pm$ 0.5% RH/year
<b>Sensing period</b>	Average: 2s
<b>Interchangeability</b>	<i>fully interchangeable</i>
<b>Dimensions size</b>	12*15.5*5.5mm

Tabel 3 menjelaskan cara kerja dari sensor kelembaban udara/*Humidity* DHT11 yang memiliki empat buah kaki, yaitu pada bagian kaki ( $V_{CC}$ ), dihubungkan ke bagian  $V_{SS}$  yg bernilai sebesar 5V, pada *board* arduino uno dan untuk bagian kaki GND dihubungkan ke *ground* (GND) pada *board arduino uno*, sedangkan pada bagian kaki

data yang merupakan keluaran (*output*) dari hasil pengolahan data *analog* dari sensor DHT11 yang dihubungkan ke bagian *analog input* (*pin 3*), yaitu pada bagian *pin* PWM (*Pulse Width Modulation*) pada *board* arduino uno terdapat satu kaki tambahan yaitu kaki NC (*Not Connected*), yang tidak dihubungkan ke *pin* manapun. Sensor kelembaban lain yang banyak dikembangkan adalah jenis sensor serat optik yang menggunakan serat optik sebagai bahan sensor.

### 3. PCB (Printed Circuit Board)



Gambar 8. Bentuk Fisik PCB

(Sumber: <http://www.anekapcb.com/pcbmaterial.html>)

Gambar 8 menjelaskan PCB (*Printed Circuit Board*). PCB adalah papan tercetak yang digunakan untuk menempatkan komponen-komponen menjadi suatu rangkaian elektronika. Jenis PCB ada yang berbentuk *double layer* dan *single layer*. PCB berjenis *double layer* memiliki dua lapisan tembaga dan yang berjenis *single layer* hanya memiliki satu lapisan tembaga. PCB mempunyai 2 jenis bahan, yaitu dari bahan pertinak dan bahan jenis fiber yang dilapisi dengan tembaga. Pada lapisan tembaga ini berfungsi sebagai penghubung komponen satu dengan yang lain sesuai rangkaian (S, Wasito. 2001).

#### 4. Radio Fekkuensi (RF) Modem



Gambar 9. KYL 200U

(Sumber: [www.aliexpress.com](http://www.aliexpress.com))

Gambar 9 adalah bentuk fisik dari KYL-200U. KYL 200U adalah jenis mikro daya nirkabel *transceiver* modul data. Dengan ukuran kecil, konsumsi daya yang rendah serta stabilitas yang baik dan kehandalan, secara luas digunakan dalam *remote control*, otomatisasi industri, telemetri nirkabel dan sebagainya. Modul ini dapat dihubungkan dengan mikrokontroler, pc, peralatan dan perangkat lain dengan *port* UART RS485 langsung.



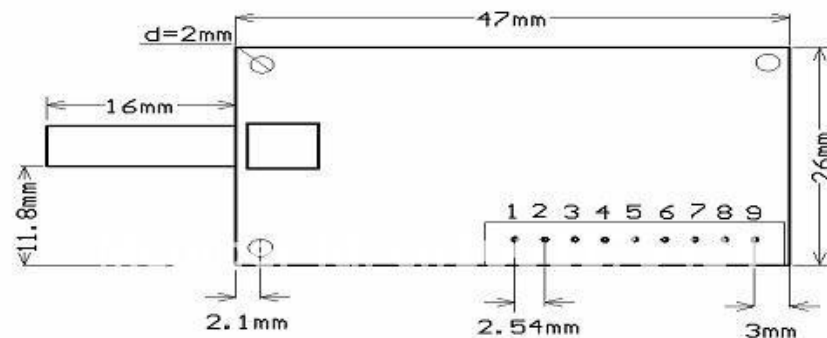
a. Spesifikasi teknis

Tabel 4. Spesifikasi Teknis pada KYL 200U

Jarak transmisi: 600 m-1000 m
Dimensi: 47x26x10mm
Berat: 22g
<i>Output</i> daya: DC 5V
Daya rf: 50 mw;
Menerima saat ini: <20mA;
Pemancar saat ini: <40mA;
Tidur saat ini: <20uA;
Menerima sensitivitas:-120dbm (1200bps);-115dbm (9600bps)
<i>Interface</i> : rs-485, rs-232, TTL (antarmuka yang dibutuhkan)
Pita frekuensi: 433 MHz (400-470 MHz tersedia)
Rf tingkat efektif: 1200,2400, 4800,9600, 19200 bps (data <i>rate</i> yang dibutuhkan)
Suhu operasi:-35 ~ + 75 Celcius (standar industri)
Modulasi: FSK/GFSK
8 saluran, diperpanjang
Stabilitas frekuensi: $\pm 2.5$ ppm industri
Antarmuka format data: 8e1/8N1 (gagal bayar)/8O1

Tabel 4 menjelaskan tentang spesifikasi teknis dari modul KYL 200U untuk mempermudah pengguna dalam menggunakan dan mengaplikasikannya pada alat yang membutuhkan modul KYL ini. Pada modul KYL ini memerlukan tegangan 5V *power supply*.

b. Definisi *pin* (9 *pin*)



Gambar 10. Dimensi Instalasi pada KYL 200U

(Sumber: [www.aliexpress.com](http://www.aliexpress.com))

Pada gambar 10 menjelaskan dimensi instalasi pada KYL 200U. Ada 9 *pin* yang terlihat dari gambar 11, definisi dari setiap *pin* tersebut akan dijelaskan pada tabel 5. Dari ke-9 *pin* tersebut, hanya 3 *pin* yang akan digunakan, yaitu *pin*1, *pin* 2 dan *pin* 3 pada penerima, sedangkan pada pengirim menggunakan *pin* 1, *pin* 2 dan *pin* 4.

Tabel 5. Definisi *Pin* pada KYL 200U

Pin no.	Nama sinyal	Fungsi	Tingkat	Koneksi dengan terminal	Komentar
1	Gnd	Landasan dari <i>power supply</i>		Tanah	
2	Vcc	<i>Power supply</i> dc	5 v		
3	Rxd/ttl	Data penerima	TTL	Txd	
4	Txd/ttl	Transmisi data	TTL	Rxd	
5	Sgnd	Sinyal			
6	A (txd)	Yang dari rs-485 (txd dari rs-232)		A (RXD)	
7	B (RXD)	B rs-485 (RXD dari rs-232)		B (txd)	
8	Tidur	Tidur control		Tidur sinyal	Tingkat rendah yang valid
9	Uji	Pengujian pabrik			

Tabel 5 menjelaskan definisi dari setiap *pin* yang ada di KYL 200U. Pin tersebut berjumlah 9. Dari 9 *pin* yang ada, yang digunakan hanya *pin* 1, *pin* 2, *pin* 3 dan *pin* 4. Sedangkan *pin* 5, *pin* 6, *pin* 7 dan *pin* 8 hanya untuk indikator dan *pin* 9 untuk uji coba dari pabrik pembuatnya. Berikut ini akan dijelaskan fungsi dari lampu indikator yang terdapat pada KYL 200U :

- 1) Indikator menyala merah untuk 0.5 detik saat dihidupkan.

- 2) Indikator berubah menjadi hijau terus saat menerima data dari udara.
  - 3) Indikator terus gelap ketika modul dalam *mode sleep*.
- c. Berikut ini fungsi kwl *wireless* untuk bidang aplikasi lainnya:
- 1) Pembacaan meter otomatis (amr);
  - 2) *Wireless alarm* dan sistem keamanan;
  - 3) Otomatisasi bangunan, sistem keamanan, memantau nirkabel;
  - 4) Transmisi data nirkabel, sistem pengumpulan data otomatis;
  - 5) Pos nirkabel, pada terminal cerdas nirkabel;
  - 6) Rf pemancar, nirkabel tampilan layar elektronik dan mesin antrian;
  - 7) Telemetry nirkabel, *remote control* dan sistem kontrol akses;
  - 8) Modem nirkabel mobil inspeksi dan empat orientasi roda;
  - 9) Sensor nirkabel, industri *remote control* nirkabel;
  - 10) Komunikasi data dalam aspek kereta api, ladang minyak, dermaga dan tentara.
  - 11) *Display* untuk petunjuk di jalan raya dan tempat umum;
  - 12) *Point to multi-point* jaringan nirkabel.

(Sumber: *RF Transceiver Module 400-470MHz 600-1000m Wireless Communication*. Diakses dari <http://id.aliexpress.com/>)

## 5. USB to TTL UART PL2303HxD



Gambar 11. USB to TTL UART PL2303HxD

(Sumber: [www.uctronics.com](http://www.uctronics.com))

Gambar 11 menjelaskan piranti **PL2303HxD** yang berguna untuk berkomunikasi dengan peralatan eksternal dengan *port* **USB** (*Universal Serial Bus*) secara *raw serial* seperti pada protokol RS-232 namun pada tingkatan tegangan sinyal yang kompatibel dengan TTL (*Transistor-Transistor Logic*), level tegangan yang umum digunakan pada UART mikrokontroler berbasis 5V. Karena fungsinya yang demikian, piranti ini dikenal dengan istilah "**USB to TTL converter**" atau "**USB to UART adapter**". Sesuai namanya, piranti ini menggunakan chip IC PL2303HX yang dikenal stabil dalam komunikasi berkecepatan tinggi. Dua buah LED SMD terpasang sebagai indikator saat transmisi aktif. Piranti ini memiliki sirkuit pelindung terhadap kondisi kelebihan arus yang berfungsi untuk menjaga peralatan dari kerusakan akibat kelebihan beban.

Piranti ini dapat digunakan sebagai modul komunikasi serial antara mikrokontroler/*development board* yang belum memiliki fitur USB

terintegrasi (misalnya Arduino Pro Mini). Pada beberapa aplikasi modul ini juga digunakan sebagai programmer untuk mengunggah kode program ke mikrokontroler yang mendukung *in-circuit serial programming* sehingga modul ini sering juga disebut sebagai "**USB UART Programmer**" walaupun pada prakteknya akan sedikit merepotkan karena harus melakukan *reset* secara manual (untuk kepentingan ini lebih baik menggunakan perangkat khusus pengunggah kode program seperti USB ASP).

(Sumber: *USB to TTL UART*. Diakses dari <http://www.vcc2gnd.com>)

### C. Perangkat Lunak (*Software*)

Pada pemrograman ini menggunakan dua program yaitu Microsoft Visual Studio 2010 dan *Arduino Compiler*. Berikut penjelasan dari kedua *software* tersebut.

#### 1. Microsoft Visual Studio 2010

Microsoft Visual Studio 2010 atau biasa disebut VB .Net 2010 merupakan salah satu aplikasi yang sering digunakan oleh para programmer yang sifatnya berbasis Windows, maka dari itu alangkah baiknya sebelum belajar lebih jauh mengenai VB .Net 2010 anda mengetahui dulu pengertian, keistimewaan dan juga sejarah dari Microsoft Visual Studio 2010.

##### a. Pengertian Microsoft Visual Studio 2010

Visual Studio 2010 pada dasarnya adalah sebuah bahasa pemrograman komputer. Dimana pengertian dari bahasa pemrograman itu adalah perintah-perintah atau instruksi yang

dimengerti oleh komputer untuk melakukan tugas-tugas tertentu. Visual Studio 2010 (yang sering juga disebut dengan VB .Net 2010) selain disebut dengan bahasa pemrograman, juga sering disebut sebagai sarana (*tool*) untuk menghasilkan program-program aplikasi berbasis Windows (Yuswanto & Subari, 2007).

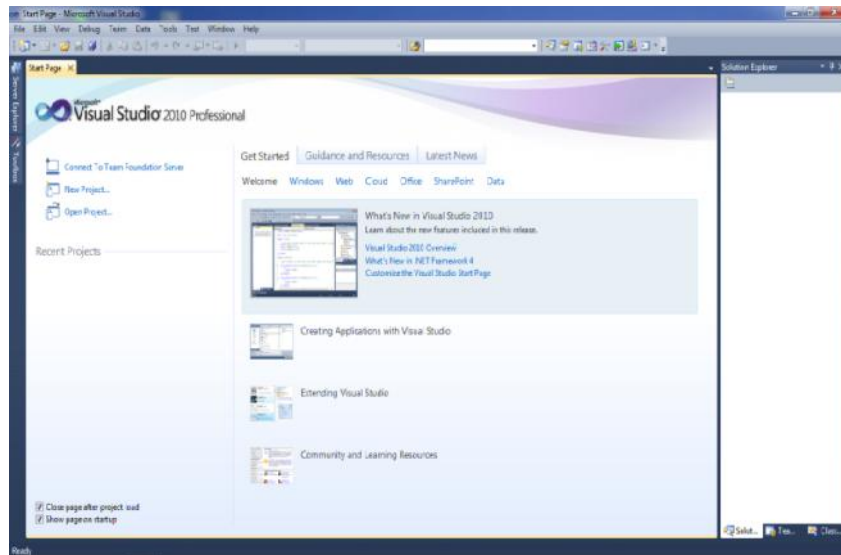
Beberapa kemampuan atau manfaat dari Visual Studio 2010 diantaranya seperti :

- 1) Untuk membuat program aplikasi berbasis Windows.
- 2) Untuk membuat objek-objek pembantu program seperti: kontrol *ActiveX*, *file Help*, aplikasi Internet dan sebagainya.
- 3) Menguji program (*debugging*) dan menghasilkan program berakhiran EXE yang bersifat *executable* atau dapat langsung dijalankan.

Pada gambar 13 menjelaskan Microsoft Visual Studio 2010. Bagi programmer pemula yang baru ingin belajar program, lingkungan Visual Studio dapat membantu membuat program dalam sekejap mata. Sedang bagi programmer tingkat lanjut, kemampuan yang besar dapat digunakan untuk membuat program-program yang kompleks, misalnya lingkungan *net-working* atau *client server*.

Bahasa Visual Studio cukup sederhana dan menggunakan kata-kata bahasa Inggris yang umum digunakan. Kita tidak perlu lagi menghafalkan sintaks-sintaks maupun format-format bahasa yang bermacam-macam, didalam Visual Basic semuanya sudah disediakan dalam pilihan-pilihan yang tinggal diambil sesuai dengan kebutuhan. Selain itu, sarana pengembangannya yang bersifat visual

memudahkan kita untuk mengembangkan aplikasi berbasis Windows, bersifat *mouse-driven* (digerakkan dengan *mouse*) dan berdaya guna tinggi.



Gambar 12. Tampilan Microsoft Visual Studio 2010

#### b. Sejarah Singkat Microsoft Visual Studio 2010

Berikut ini beberapa bagian penting dalam sejarah perkembangannya:

- 1) Pertama kali di *release* dengan nama Visual Basic yang dikeluarkan pada tahun 1991, yaitu Visual Basic yang masih berbasis DOS dan untuk Windows.
- 2) Visual Basic 3.0 dirilis pada tahun 1993.
- 3) Visual Basic 4.0 dirilis pada tahun 1994 dengan dukungan untuk aplikasi 32 *bit*.
- 4) Visual Basic 6.0 dirilis pada akhir tahun 1998.
- 5) Visual Basic untuk selanjutnya yaitu versi .Net yang dirilis awal 2002.



- 6) Pada tahun 2003 Visual Studio 2003 dirilis untuk memperbaiki kinerja dari Visual Studio 2002 dengan meluncurkan .NET Framework versi 1.1.
- 7) Pada tahun 2005 Microsoft mengeluarkan Visual Basic Versi 8.0 atau biasa disebut Microsoft Visual Studio 2005.
- 8) Pada tahun 2008 Microsoft juga mengeluarkan versi 9.0 atau biasa disebut Microsoft Visual Studio 2008.
- 9) Hingga pada tahun 2010 Visual Studio sudah pada versi 10.0 atau biasa disebut Microsoft Visual Studio 2010, dimana didalamnya sudah ditambahkan bahasa pemrograman yang baru yaitu F# yang merupakan penyempurnaan dari versi-versi sebelumnya.

**c. Keistimewaan Microsoft Visual Studio 2010**

Beberapa keistimewaan Visual Studio 2010 ini diantaranya seperti :

- 1) Menggunakan *platform* pembuatan program yang dinamakan *developer studio*, yang memiliki tampilan dan sarana yang sama dengan Visual C++ dan Visual J++. Dengan begitu dapat bermigrasi atau belajar bahasa pemrograman lainnya dengan mudah dan cepat.
- 2) Memiliki *compiler* handal yang dapat menghasilkan *file executable* yang lebih cepat dan lebih efisien dari yang sebelumnya.
- 3) Memiliki beberapa tambahan *wizard* yang baru. *Wizard* adalah sarana yang mempermudah didalam pembuatan aplikasi dengan mengotomisasi tugas-tugas tertentu.
- 4) Visual Studio 2010 mempunyai beberapa fitur untuk pengembangan berbagai macam aplikasi yang diantaranya;

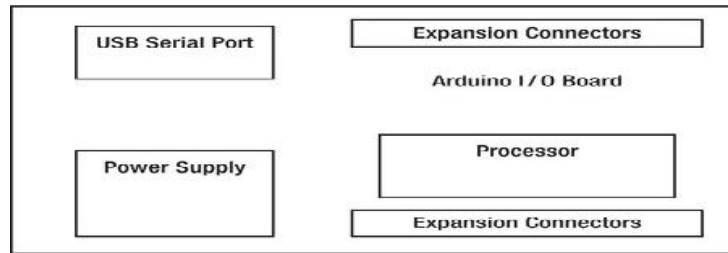
Windows Development, Web Development, Office Development, Sharepoint Development, Cloud Development (Windows Azure), Silverlight Tooling, Multi-Core Development, Customizable IDE.

## 2. Arduino Compiler

Arduino adalah *platform* pembuatan *prototype* elektronik yang bersifat *open-source hardware* yang berdasarkan pada perangkat keras dan perangkat lunak yang fleksibel dan mudah digunakan. Arduino ditujukan bagi para seniman, desainer, dan siapapun yang tertarik dalam menciptakan objek atau lingkungan yang interaktif (Djuandi, Feri. 2011).

Arduino pada awalnya dikembangkan di Ivrea, Italia. Nama Arduino adalah sebuah nama maskulin yang berarti teman yang kuat. *Platform* arduino terdiri dari arduino *board*, *shield*, bahasa pemrograman arduino, dan arduino *development environment*. Arduino *board* biasanya memiliki sebuah chip dasar mikrokontroler Atmel AVR ATmega8 berikut turunannya.

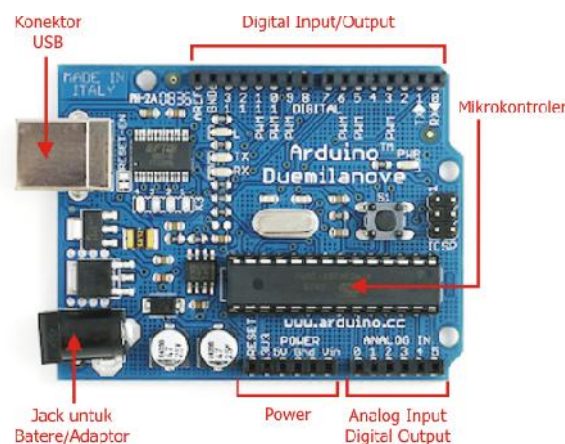
Blok diagram arduino *board* yang sudah disederhanakan dapat dilihat pada gambar 13. *Shield* adalah sebuah papan yang dapat dipasang diatas arduino *board* untuk menambah kemampuan dari arduino *board*. Bahasa pemrograman arduino adalah bahasa pemrograman yang umum digunakan untuk membuat perangkat lunak yang ditanamkan pada arduino *board*. Bahasa pemrograman arduino mirip dengan bahasa pemrograman C++.



Gambar 13. Blok Diagram pada *board* arduino

Sumber: <http://www.arduino.cc>

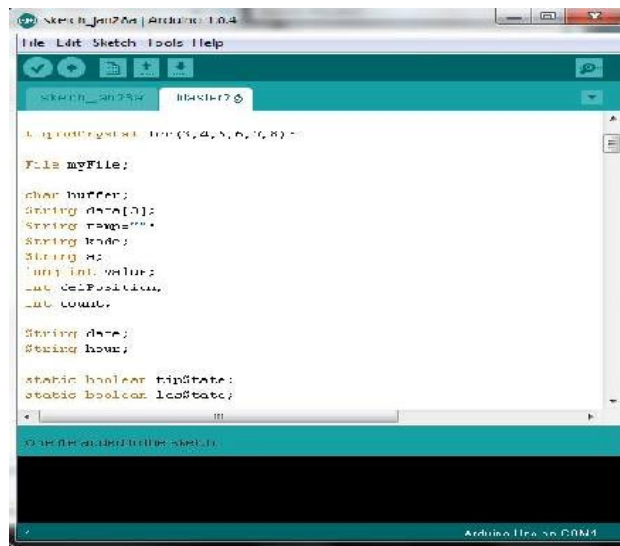
Arduino uno merupakan salah satu jenis rangkaian mikrokontroller yang menggunakan sistem *physical computing*. *Physical computing* adalah membuat sebuah sistem atau perangkat fisik dengan menggunakan *software* dan *hardware* yang sifatnya interaktif yaitu dapat menerima rangsangan dari lingkungan dan merespon balik. *Physical computing* adalah sebuah konsep untuk memahami hubungan yang manusiawi antara lingkungan yang sifat alaminya adalah analog dengan dunia digital (Djuandi,2011). Supaya lebih jelas bentuk dari arduino uno dapat dilihat pada gambar 14.



Gambar 14. *Board* Arduino

Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi ia adalah kombinasi dari *hardware*, bahasa pemrograman dan ***Integrated***

**Development Environment (IDE)** yang canggih. IDE adalah sebuah *software* yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam *memory* mikrokontroller. Ada banyak proyek dan alat-alat dikembangkan oleh akademisi dan profesional dengan menggunakan Arduino, selain itu juga ada banyak modul-modul pendukung (sensor, tampilan, penggerak dan sebagainya) yang dibuat oleh pihak lain untuk bisa disambungkan dengan Arduino. Supaya lebih jelas tampilan pada IDE dapat dilihat pada gambar 15.



Gambar 15. *Interface* dari Arduino IDE

Pada umumnya *sketch* yang dibuat di *Arduino Software* di-*compile* dengan perintah *verify/Compile* (Ctrl+R) lalu hasilnya di-*download* ke papan Arduino seperti Arduino Uno atau Arduino Mega 2560. Program hasil kompilasi itu dijalankan oleh *bootloader*.

Arduino Uno adalah *arduino board* yang menggunakan mikrokontroler ATmega328. Arduino Uno memiliki 14 *pin* digital (6 *pin* dapat digunakan sebagai *output* PWM), 6 *input* analog, sebuah 16 MHz osilator kristal, sebuah koneksi USB, sebuah konektor sumber tegangan,

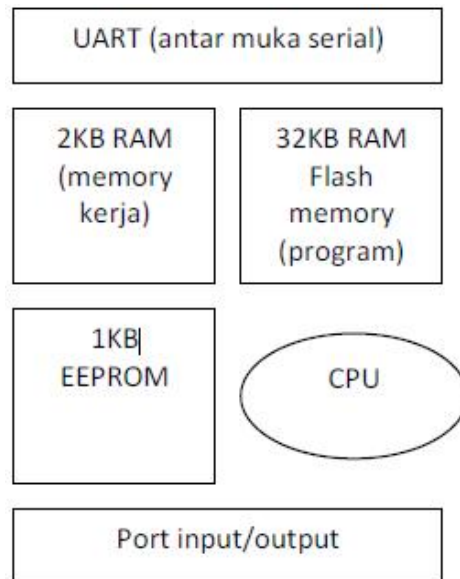
sebuah *header* ICSP, dan sebuah tombol *reset*. Arduino Uno memiliki area cakupan yang luas untuk segala hal yang dibutuhkan untuk mendukung sebuah aplikasi yang berbasis mikrokontroler. Hanya dengan menghubungkannya ke sebuah komputer melalui USB atau memberikan tegangan DC dari baterai atau adaptor AC ke DC sudah dapat membuat aplikasinya bekerja. Arduino Uno menggunakan ATmega16U2 yang diprogram sebagai *USB-to-serial converter* untuk komunikasi serial ke komputer melalui port USB.

Adapun data teknis yang terdapat board Arduino UNO R3 adalah sebagai berikut:

- Mikrokontroler: ATmega328
- Tegangan Operasi: 5V
- Tegangan *Input (recommended)*: 7 - 12 V
- Tegangan *Input (limit)*: 6-20 V
- *Pin* digital I/O: 14 (6 diantaranya *pin* PWM)
- *Pin* Analog *input*: 6 *input pin*
- Arus DC per *pin* I/O: 40 mA
- Arus DC untuk *pin* 3.3 V: 150 mA
- *Flash Memory*: 32 KB dengan 0.5 KB digunakan sebagai *bootloader*
- SRAM: 2 KB
- EEPROM: 1 KB
- Kecepatan besaran waktu sebesar: 16 Mhz sebagai komponen untuk (*Crystall oscillator*)

Berikut ini akan dijelaskan gambaran mengenai apa saja yang terdapat di dalam sebuah mikrokontroller dan diagram blok sederhana

dari mikrokontroler ATmega328 yang dipakai pada Arduino Uno yang dapat dilihat pada gambar 16:



Gambar 16. Diagram Blok Sederhana dari ATmega328

(Sumber: <http://www.arduino.cc>)

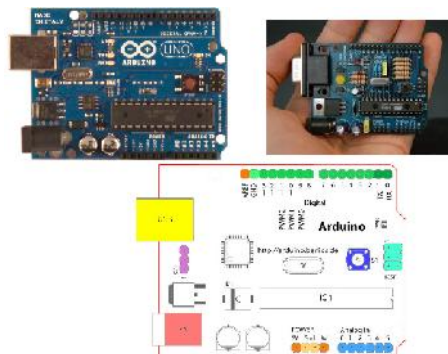
Blok-blok pada gambar 16 dijelaskan sebagai berikut:

- *Universal Asynchronous Receiver/Transmitter* (UART) adalah antar muka yang digunakan untuk komunikasi serial seperti pada RS-232, RS-422 dan RS-485.
- 2KB RAM pada *memory* kerja bersifat *volatile* (hilang saat daya dimatikan), digunakan oleh variable-variabel di dalam program.
- 32KB RAM *flash memory* bersifat *non-volatile* yang digunakan untuk menyimpan program yang dimuat dari komputer. Selain program, *flash memory* juga menyimpan *bootloader*. *Bootloader* ini yang menjembatani antara *software compiler* arduino dengan mikrokontroler, sehingga pengguna papan mikrokontroler arduino pada saat menulis program

tidak perlu banyak menuliskan sintak bahasa C, dan cukup melakukan pemanggilan fungsi program.

- 1KB EEPROM bersifat *non-volatile*, digunakan untuk menyimpan data yang tidak boleh hilang saat daya dimatikan.
- *Central Processing Unit (CPU)*, bagian dari mikrokontroler yang berfungsi untuk menjalankan setiap instruksi dari program.
- *Port input/output*, pin-pin yang berfungsi untuk menerima data (*input*) digital atau analog, dan mengeluarkan data (*output*) digital atau analog.

Setelah mengenal bagian-bagian utama dari mikrokontroler ATmega328 sebagai komponen utama, selanjutnya kita akan mengenal bagian-bagian dari papan Arduino itu sendiri. Diambil contoh sebuah papan Arduino tipe USB dan bagian-bagiannya dapat dijelaskan sebagai berikut seperti pada gambar 17:



Gambar 17. Bagian-bagian pada papan Arduino Uno

(Sumber: <http://arduino.cc/en/main/hardware>)

Tabel 6. Tabel karakteristik rangkaian pada *board* arduino uno

<p><b>14 pin input/output digital (0-13)</b></p> <p>Berfungsi sebagai <i>input</i> atau <i>output</i>, dapat diatur oleh program.</p> <p>Khusus untuk 6 buah <i>pin</i> 3, 5, 6, 9, 10 dan 11, dapat juga berfungsi sebagai <i>pin</i> analog <i>output</i> dimana tegangan <i>output</i>-nya dapat diatur. Nilai sebuah <i>pin output</i> analog dapat diprogram antara 0 – 255, dimana hal itu mewakili nilai tegangan 0 – 5V</p>
<p><b>USB</b></p> <p>Berfungsi untuk:</p> <ul style="list-style-type: none"> <li>- Memuat program dari komputer ke dalam papan</li> <li>- Komunikasi serial antara papan dan komputer</li> <li>- Memberi daya listrik kepada papan</li> </ul>
<p><b>Sambungan SV1</b></p> <p>Sambungan atau <i>jumper</i> untuk memilih sumber daya papan, apakah dari sumber eksternal atau menggunakan USB. Sambungan ini tidak diperlukan lagi pada papan Arduino versi terakhir karena pemilihan sumber daya eksternal atau USB dilakukan secara otomatis</p>
<p><b>Q1 – Kristal (<i>quartz crystal oscillator</i>)</b></p> <p>Jika mikrokontroller dianggap sebagai sebuah otak, maka kristal adalah jantung-nya karena komponen ini menghasilkan detak-detak yang dikirim kepada mikrokontroller agar melakukan sebuah operasi untuk setiap detak-nya. Kristal ini dipilih yang berdetak 16 juta kali per detik (16MHz).</p>



lanjutan tabel 6...

<b>Tombol Reset S1</b>  Untuk me- <i>reset</i> papan sehingga program akan mulai lagi dari awal. Perhatikan bahwa tombol <i>reset</i> ini bukan untuk menghapus program atau mengosongkan mikrokontroler.
<b><i>In-Circuit Serial Programming (ICSP)</i></b>  <i>Port</i> ICSP memungkinkan pengguna untuk memprogram mikrokontroler secara langsung, tanpa melalui <i>bootloader</i> . Umumnya pengguna Arduino tidak melakukan ini sehingga ICSP tidak terlalu dipakai walaupun disediakan.
<b>IC 1 – Mikrokontroler ATmega</b>  Komponen utama dari papan Arduino, di dalamnya terdapat CPU, ROM dan RAM.
<b>sum X1 – sumber</b>  Jika hendak disuplai dengan sumber daya eksternal, papan Arduino dapat diberikan tegangan DC antara 9-12V.
<b>6 pin input analog (0-5)</b>  Pin ini sangat berguna untuk membaca tegangan yang dihasilkan oleh sensor analog, seperti sensor suhu. Program dapat membaca nilai sebuah <i>pin input</i> antara 0 – 1023, dimana hal itu mewakili nilai tegangan 0 – 5V.

Pada tabel 6 menjelaskan tentang karakteristik rangkaian pada *board* arduino. Tanpa melakukan konfigurasi apapun, begitu sebuah papan arduino yang baru dikeluarkan dari kotak pembungkusnya dapat langsung

disambungkan ke sebuah komputer melalui kabel USB. Selain berfungsi sebagai penghubung untuk pertukaran data, kabel USB ini juga akan mengalirkan arus DC sebesar 5 Volt ke papan arduino sehingga praktis tidak diperlukan sumber daya dari luar. Saat mendapat suplai daya, lampu LED indikator daya pada papan Arduino akan menyala yang menandakan bahwa papan Arduino siap bekerja.

Pada papan Arduino Uno terdapat sebuah LED kecil yang terhubung ke *pin* digital 13. LED ini dapat digunakan sebagai *output* saat seorang pengguna membuat sebuah program dan ia membutuhkan sebuah penanda dari jalannya program tersebut. Hal ini adalah cara yang praktis saat pengguna melakukan uji coba.

Umumnya mikrokontroller pada papan arduino telah memuat sebuah program kecil yang akan menyalakan LED tersebut dan akan berkedip-kedip dalam jeda satu detik. Jadi, sangatlah mudah untuk menguji apakah sebuah papan arduino baru dalam kondisi baik atau tidak, yaitu cukup dengan menyambungkan papan arduino itu dengan sebuah komputer dan perhatikan apakah LED indikator daya menyala konstan dan LED dengan *pin*-13 itu menyala berkedip-kedip. Supaya lebih jelas bisa dilihat pada gambar 19.



Gambar 18. Komponen LED yang terdapat pada *board* Arduino Uno

(Sumber: <http://arduino.cc/en/main/hardware>)

#### D. Pengertian Suhu dan Kelembaban

##### 1. Pengertian Suhu

Suhu merupakan sifat tanah yang amat penting, karena dapat mempengaruhi pertumbuhan tanaman secara langsung dan juga mempengaruhi lengas, aerasi, struktur, kegiatan mikrobia dan enzim, perombakan sisa-sisa tanaman, dan ketersediaan zat-zat hara tanaman. Suhu merupakan salah satu faktor pertumbuhan yang penting bagi tanaman, sebagai mana layaknya air, udara, atau zat-zat hara mineral. Biji, akar tanaman, dan mikrobia yang tumbuh di dalam tanah, dan proses kehidupan mereka secara langsung dipengaruhi oleh suhu tanah (Kanginan, Marthen. 2004).

Pengaruh penting suhu terhadap pertumbuhan tanaman terutama melalui lengas tanah. Aerasi tanah dipengaruhi oleh perbedaan-perbedaan suhu dan kandungan lengas. Pengaruh suhu terhadap struktur tanah yakni melalui pembekuan dan pencairan. Suhu tanah baik semata-

mata oleh peningkatan atau penurunan atau oleh pembekuan air tanah, memiliki pengaruh nyata terhadap perombakan bahan organik dan mineral tanah, pembebasan unsur hara tanaman dan juga terhadap pembentukan lempung. Mungkin kita masih ingat bahwa reaksi kimia akan meningkat sebesar dua kali lipat dengan setiap kenaikan suhu sebesar 10°C. Dalam klasifikasi tanah dunia sering kali juga mengikutkan faktor suhu sebagai salah satu kriteria pembedanya.

## **2. Pengertian Kelembaban**

Kelembaban udara adalah banyaknya kandungan uap air di atmosfer. Udara atmosfer adalah campuran dari udara kering dan uap air. Kelembaban udara merupakan tingkat kebasahan udara karena dalam udara air selalu terkandung dalam bentuk uap air. Uap air adalah suatu gas yang tidak dapat dilihat, yang merupakan salah satu bagian dari atmosfer. Kabut dan awan adalah titik air atau butir-butir air yang melayang-layang di udara. Kabut melayang-layang dekat permukaan tanah, kalau awan melayang-layang di angkasa. Banyaknya uap air yang dikandung oleh hawa tergantung pada temperatur. Makin tinggi temperatur makin banyak uap air yang dapat dikandung oleh hawa (Hardjodinomo,1975).

Seperti gas-gas lainnya, uap air juga mempunyai tekanan yang makin lebih besar apabila temperatur naik. Tekanan tersebut dinamakan tekanan uap. Tekanan uap adalah tekanan yang diberikan atau ditimbulkan oleh uap air sebagai bagian dari udara pada temperatur yang tertentu. Tekanan uap merupakan bagian dari tekanan udara yang dapat diukur dengan milimeter air raksa atau milibar. Jika udara pada suatu

temperatur sudah kenyang (jenuh) maka tekanan uap pada temperatur tersebut mencapai batas maksimum. Angka maksimum tersebut disebut tekanan uap maksimum (Zailani, 1986). Proses perubahan air menjadi uap air disebut penguapan (vaporisasi atau evaporasi). Molekul-molekul air yang mempunyai energi kinetik yang cukup untuk mengatasi gaya-gaya tarik yang cenderung untuk menahannya dalam badan air diproyeksikan melalui permukaan air. Oleh karena energi kinetik bertambah dan tegangan permukaan berkurang ketika temperatur naik, maka laju permukaan naik menurut temperatur. Hampir semua uap di atmosfer adalah hasil penguapan dari permukaan air (Linsley, 1989).

Beberapa prinsip umum yang digunakan dalam pengukuran kelembaban udara yaitu:

- a. Metode pertambahan panjang
- b. Berat, pada benda-benda hidroskopis
- c. Metode termodinamika

Alat pengukur kelembaban udara secara umum disebut higrometer, sedangkan yang menggunakan metode termodinamika disebut psikrometer (Kartasapoetra, 1990). Berikut ini adalah macam-macam kelembaban udara:

- a. Kelembaban relatif atau nisbi, yaitu perbandingan jumlah uap air di udara dengan yang terkandung di udara dengan suhu yang sama.
- b. Kelembaban absolut atau mutlak, yaitu banyaknya uap air pada gram dalam 1m<sup>3</sup>.

Beberapa cara untuk menyatakan jumlah uap air, yaitu:

- a. Tekanan uap adalah tekanan parsial dari uap air. Dalam fase gas,

maka uap air di dalam atmosfer seperti gas sempurna (ideal).

- b. Kelembaban mutlak yaitu massa air yang terkandung dalam satu satuan volume udara lengas.
- c. Nisbah percampuran (mixing ratio) yaitu nisbah massa uap air terhadap massa udara kering.
- d. Kelembaban spesifik didefinisikan sebagai massa uap air persatuan massa uap air basah.
- e. Kelembaban nisbi (RH) ialah perbandingan nisbah percampuran dengan nilai jenuhnya dan dinyatakan dalam %.
- f. Suhu virtual.

Besaran yang sering dipakai untuk menyatakan kelembaban udara adalah kelembaban nisbi yang diukur dengan psikrometer atau higrometer. Kelembaban nisbi dapat berubah sesuai tempat dan waktu. Pada siang hari kelembaban nisbi berangsur-angsur turun kemudian pada sore hari sampai menjelang pagi bertambah besar.

## **BAB III**

### **KONSEP RANCANGAN**

Perancangan *Prototype* Telemetri Alat Pendeteksi Dini Kebakaran Hutan Menggunakan ATmega8 dengan Antarmuka Komputer. Secara berurutan terdiri dari proses identifikasi kebutuhan, analisis kebutuhan, perancangan perangkat keras (*hardware*), perancangan perangkat lunak (*software*), pembuatan alat serta pengujiannya.

#### **A. IDENTIFIKASI KEBUTUHAN**

Dalam pembuatan proyek akhir ini dapat dilakukan beberapa identifikasi kebutuhan adalah sebagai berikut:

1. Rangkaian sistem minimum mikrokontroller ATmega8 sebagai pengontrol seluruh kinerja dari alat pendeteksi dini kebakaran hutan.
2. Dibutuhkan bahasa pemrograman yang sesuai untuk mengaplikasikan algoritma dari alat pendeteksi dini kebakaran hutan.
3. RF Modem KYL 200U sebagai media pengirim dan penerimanya.
4. Sensor DHT11 sebagai pendeteksi dari perubahan suhu dan kelembaban.
5. Menggunakan Arduino kompiler untuk memprogram sensor DHT11.
6. Komputer/Laptop sebagai media antarmukanya.
7. Dibutuhkannya sumber tegangan dari adaptor 12V untuk menjalankan alat.
8. *Box* plastik sebagai *box* pelindung komponen.

#### **B. Analisa Kebutuhan**

Bedasarkan identifikasi kebutuhan di atas, maka diperoleh beberapa analisis kebutuhan terhadap sistem yang akan dirancang, diantaranya:

1. Rangkaian sistem minimum mikrokontroler menggunakan ATmega8 membutuhkan beberapa komponen pendukung diantaranya rangkaian penurun tegangan untuk memberikan suplai tegangan yang dibutuhkan mikrokontroler sebesar 5V, tombol *reset* untuk mereset mikrokontroller, dan beberapa *output* yang meliputi sensor DHT11 dan RF Modem KYL 200U.
2. Menggunakan Visual Basic untuk membuat *interface* dari *prototype* telemetri alat pendeteksi dini kebakaran hutan. Pada *interface* yang akan dibuat, terdapat beberapa bagian, diantaranya pilihan *default port* yang digunakan untuk menghubungkan USB ke komputer/laptop, pilihan *baudrate* untuk menentukan tingkat sinyal yang dibutuhkan, tombol *connect* dan *disconnect* untuk menghubungkan dan memutuskan koneksi antar RF Modem, dan tampilan hasil pembacaan dari suhu, kelembaban, status, dan indikator koneksi dari usb.
3. Menggunakan RF Modem KYL 200U sebagai *transceiver* untuk mengirimkan dan menerima data. RF Modem KYL 200U ini terdiri dari dua bagian, yaitu *receiver* dan *transmitter*. Pada *receiver* terpasang USB to TTL yang digunakan untuk menghubungkan *interface* dengan alat. Sebelum *transceiver* digunakan, dipasang driver PL 2303 agar saat USB dihubungkan dapat membaca *port* COM yang terdeteksi. Sedangkan pada *transmitter*, terdapat beberapa bagian diantaranya sistem minimum, sensor DHT11, saklar, dan terminal untuk adaptor 12V.
4. Sensor suhu dan kelembaban menggunakan sensor DHT11 yang terpasang pada *transceiver* modul KYL 200U. Hasil pengukuran suhu dan



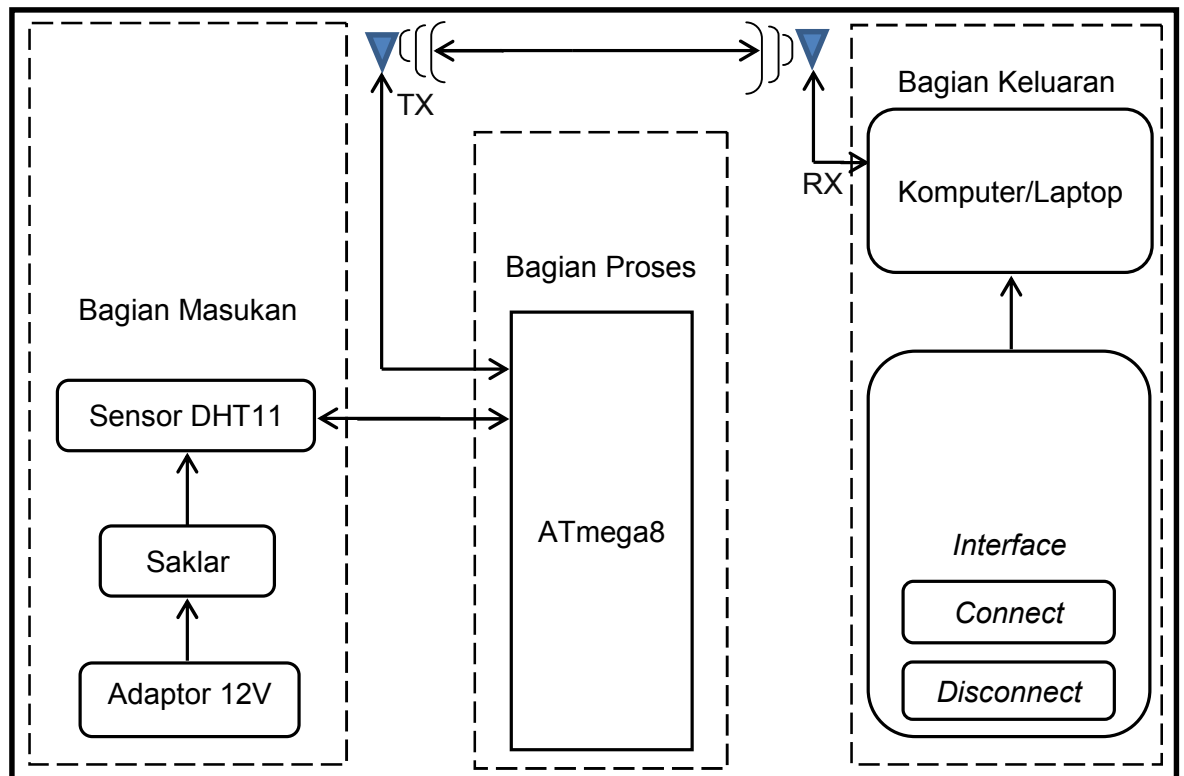
kelembaban yang terbaca kemudian akan dikirimkan ke *receiver* pada modul KYL 200U.

5. Menggunakan Arduino Compiler untuk mengkompile bahasa C arduino dan juga untuk mengunggah program hasil kompilasi tersebut (*hex file*) ke modul Arduino.
6. Media antarmuka menggunakan komputer/laptop sebagai perangkat untuk membangun *interface* pembacaan suhu dan kelembaban. Pada *interface* ini terdapat pilihan *default port* USB untuk menghubungkan *receiver* ke komputer/laptop, selain itu terdapat pula pilihan *baudrate* (jumlah kali sinyal per detik) yang diatur sesuai standar karena apabila terlalu rendah maka kecepatan pengiriman akan lambat tetapi tingkat error rendah begitupula sebaliknya apabila diatur semakin tinggi maka pengiriman akan semakin cepat tetapi tingkat error akan semakin tinggi, tombol *connect* dan *disconnect* untuk menghubungkan maupun memutuskan koneksi antara *receiver* dengan *transmitter* dan *datalog* yang otomatis tersimpan di *drive* D yang merupakan rekaman data hasil pengukuran suhu dan kelembaban.
7. Menggunakan adaptor yang bertegangan 12V sebagai sumber tegangan pada transceiver untuk menjalankan sensor DHT11 dan RF Modem KYL 200U. Pada alat pendeteksi ini membutuhkan tegangan antara 4,5V – 5,5V supaya mikrokontroler dapat berjalan tetapi tidak boleh lebih dari 12V karena dapat menyebabkan IC regulator menjadi terlalu panas.
8. Sebagai tempat dari alat ini menggunakan *box* plastik, agar terlihat rapi. Dibutuhkan 2 *box* plastik, *box* plastik yang pertama untuk *receiver* yang berukuran 7.5cm x 5cm. *Box* plastik tersebut sudah cukup untuk menampung usb to tll dan *receiver* dari RF Modem KYL 200U. Sedangkan

*box* plastik yang kedua untuk *transmitter* menggunakan *box* plastik besar yang berukuran 14.5cm x 9.5cm yang digunakan untuk menampung *transmitter* dari RF Modem KYL 200U, sistem minimum dari ATmega8 dan sensor DHT11. Kedua *box* plastik ini dapat dibeli di toko elektronik.

### C. Perancangan Alat

#### 1. Perancangan Alat Keras (*Hardware*)



Gambar 19. Blok Diagram Rangkaian

Gambar 19 adalah blok diagram dari *hardware* alat pendeteksi dini kebakaran hutan yang akan dibuat. Dibawah ini akan dijelaskan setiap blok dari blok diagram tersebut.

### a. Blok Masukan

Blok masukan terdiri dari sistem minimum ATmega8, sensor DHT11, terminal untuk adaptor 12V dan saklar. Saat dalam keadaan *on*, lampu indikator led warna biru akan menyala dan indikator lampu led berwarna hijau yang akan berkedip saat alat siap digunakan. Sensor ini akan bekerja setelah saklar dalam posisi *on*. Sensor DHT11 ini saat akan mengalami sedikit *loading* sesaat  $\pm 10$  detik untuk membaca *file hex*. Pada bagian masukan ini terpasang saklar yang berfungsi untuk menghubungkan dan memutuskan sumber dari adaptor sebesar 12V.

### b. Blok Proses

Blok pemroses terdapat sistem minimum ATmega8 yang digunakan sebagai pemroses dan pengendali utama dari seluruh sistem.

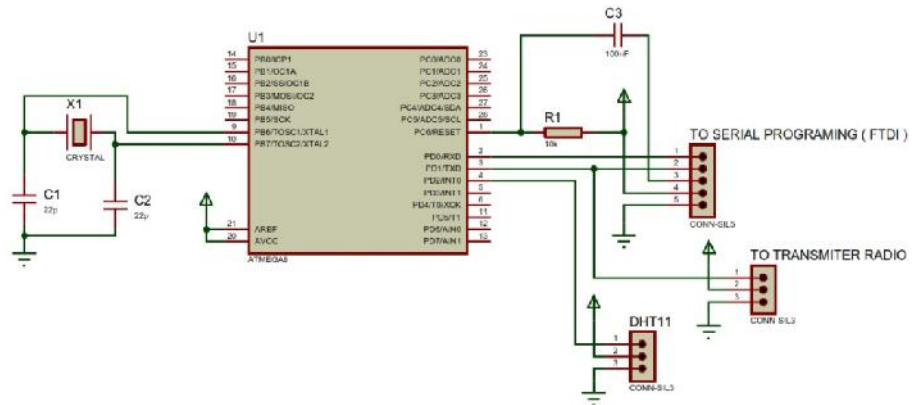
### c. Blok Keluaran

Pada blok keluaran terdapat komputer/laptop yang digunakan untuk menampilkan data suhu dan kelembaban yang terukur. Pada *interface* terdapat beberapa tampilan seperti pemilihan *default port*, pemilihan *baudrate*, tombol *connect*, dan tombol *disconnect* yang terdapat pada *interface* yang ditampilkan di komputer/laptop.

## 2. Rancangan Rangkaian

Rancangan rangkaian terdiri dari rangkaian sistem minimum ATmega8, rangkaian penurun tegangan dan rangkaian USB to TTL.

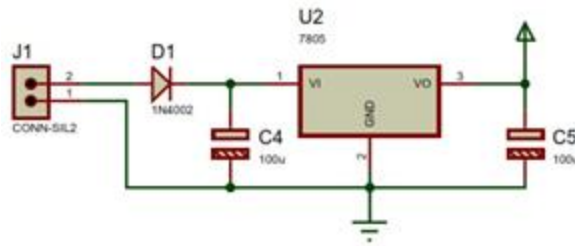
### a. Rangkaian Sistem Minimum ATmega8



Gambar 20. Rangkaian Sistem Minimum ATmega8

Gambar 20 adalah rangkaian sistem minimum yang menggunakan IC ATmega8. IC mikrokontroler ini memiliki 28 pin dengan 23 pin I/O, 8 kbyte *flash* memori, dan 512 byte EEPROM. Pin yang digunakan untuk alat ini adalah pin 1 yang khusus untuk *reset*, pin 2 untuk RX (pin yang tersedia pada sensor DHT11), pin 3 untuk TX (pin yang tersedia untuk sensor DHT11), pin 4 adalah pin *interrupt* (yaitu proses mendahulukan program di *interrupt* dan menghentikan program utama apabila terjadi interupsi, interupsi ada dua yaitu interupsi internal dan eksternal, pada sensor DHT11 menggunakan *interrupt* eksternal karena untuk membaca data dari luar), pin 9 dan 10 adalah pin khusus untuk crystal, pin 20 dan 21 adalah pin untuk *power*. Pada pin 1, 9 dan 10 terdapat rangkaian standar pada arduino, komponen yang didalamnya ialah X1 (*crystal*) sebesar 16MHz, C1 dan C2 ialah kapasitor pilar sebesar 22pF, C3 ialah kapasitor pilar sebesar 100pF, dan R1 ialah resistor sebesar 10K. Gambar rangkaian sistem minimum terlihat pada Gambar 19.

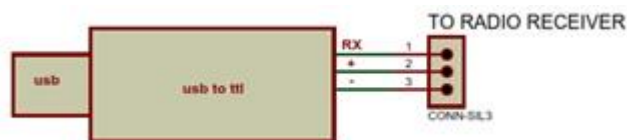
### b. Rangkaian Penurun Tegangan



Gambar 21. Rangkaian Penurun Tegangan

Rangkaian pada gambar 21 merupakan rangkaian standar pada penurun tegangan. Nilai besaran komponen C4 dan C5 bisa dinaikkan, apabila bernilai semakin besar maka akan semakin lama waktu penyimpanan tegangannya begitu pula sebaliknya. IC 7805 ini memiliki fungsi untuk menurunkan tegangan dari sumber adaptor 12V menjadi 5V, dan tegangan 5V yang dikeluarkan ini untuk mengalir arus ke sensor DHT11 dan ke *transmitter* KYL 200U. Sensor DHT11 ini akan *loading* sekitar 10 detik untuk pembacaan *file hex* pada sensor DHT11 tersebut.

### c. Rangkaian USB to TTL



Gambar 22. Rangkaian USB to TTL

Gambar 22 adalah rangkaian USB to TTL yang dapat dibeli dipasaran. USB to TTL ini akan dihubungkan dengan *receiver* pada KYL 200U yang berfungsi untuk menerima data dan menghubungkannya ke komputer/laptop lewat USB.

### 3. Perancangan Perangkat Lunak (*Software*)

Sebelumnya mikrokontroler ATmega8 ini harus terisi program sebagai pengendali dari kerja alat untuk menjalankan alat pendeteksi dini kebakaran hutan. Program yang digunakan adalah Microsoft Visual Studio 2010 (Visual Basic) untuk membuat *interface* dari media antarmuka alat pendeteksi dini kebakaran hutan.

#### a. Algoritma

##### 1. Algoritma *Flowchart* Penerima Data dari RX:

- a) *Start*
- b) Menginisialisasi serial COM 9600
- c) Menginisialisasi *timer* 1 detik
- d) Membaca serial COM
- e) *Scan* kelembaban
- f) *Scan* temperature
- g) Buat *file txt*
- h) Simpan hasil baca waktu dan tanggal di *file txt*
- i) Apakah temperature  $\geq 50$ . Jika “ya” maka tampilkan status “Warning!”. Jika “tidak” tampilkan status “aman”
- j) Tunda *timer* 1 detik untuk memproses hasil data yang terbaca
- k) Selesai

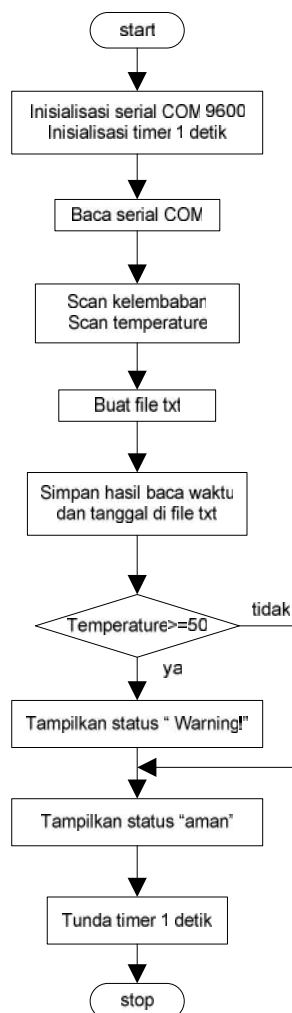
##### 2. Algoritma *Flowchart* TX

- a) *Start*
- b) Menginisialisasi serial
- c) Mengatur *baudrate* 9600
- d) Menginisialisasi DHT11

- e) Membaca kelembaban
- f) Membaca temperature
- g) Mengirim serial kelembaban
- h) Mengirim serial temperature
- i) Menunda waktu selama 1000ms untuk pembacaan *file hex* pada sensor DHT11
- j) Selesai

b. Perancangan Diagram Alir (*Flowchart*)

1. *Flowchart* Penerima Data dari RX

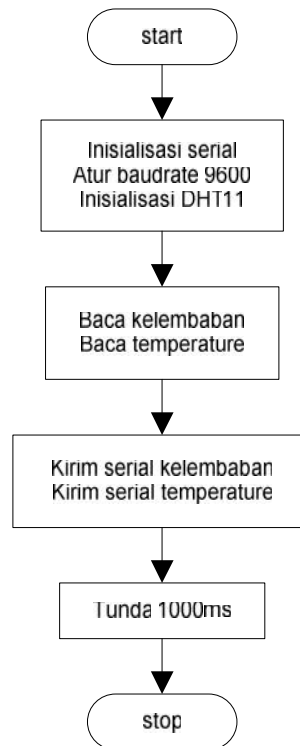


Gambar 23. *Flowchart* pada RX

Gambar 23 adalah gambar *flowchart* pada RX (penerima). Saat *start* akan menginisialisasi serial COM pada *port* usb di komputer, pada angka 9600 itu maksudnya *baudrate* (jumlah kali sinyal per detik) yaitu sebesar 9600 yang merupakan jumlah standar pengaturan yang digunakan karena apabila terlalu rendah maka kecepatan pengiriman akan lambat tetapi tingkat eror rendah begitupula sebaliknya apabila diatur semakin tinggi maka pengiriman akan semakin cepat tetapi tingkat eror akan tinggi, setelah proses inisialisasi selesai akan ada penundaan waktu 1 detik untuk membaca serial COM, setelah pembacaan serial selesai kemudian mendeteksi suhu dan kelembaban yang dikirimkan dan nilai data yang terbaca akan tercatat kemudian diteruskan untuk membuat *file txt* untuk menyimpan data ke *data log* yang berisi data kelembaban, suhu, waktu, tanggal dan status keamanan setelah itu akan mendeteksi apakah suhu  $\geq 50^{\circ}\text{C}$ , jika “ya” akan ditampilkan status “*warning*” atau jika “tidak” akan ditampilkan status “aman” dan akan ada jeda 1 detik untuk penerimaan data kemudian secara logika proses ini akan *looping* sampai ada perintah untuk berhenti, akan tetapi pada *flowchart* diatas menurut algoritma pada program akan ada proses “*stop*”.



## 2. Flowchart TX



Gambar 24. Flowchart pada TX

Gambar 24 adalah gambar *flowchart* dari TX (pengirim). Saat *start* akan menginisialisasi *port serial* yang diubahkan dari mikroprosesor ATmega8 yang akan dikonversikan ke *port* USB komputer, pengaturan dari *baudrate* yaitu sebesar 9600 dan menginisialisasi data sensor DHT11 yang akan dikirimkan, kemudian akan ada proses pembacaan suhu dan kelembaban yang akan dikirimkan ke penerima, kemudian jeda 1 detik untuk proses pengiriman. Proses ini akan terus berjalan selama alat dihidupkan.

## D. Alat dan Bahan Yang Digunakan

### 1. Alat

Dalam pembuatan alat pada proyek akhir ini menggunakan beberapa alat, diantaranya:

- a. Laptop/Komputer
- b. Multimeter
- c. Bor
- d. Setrika
- e. Pelarut PCB (*ferrichloride*)
- f. Timah
- g. Solder
- h. Atraktor
- i. Cutter
- j. Obeng

### 2. Bahan

- 1) Komponen untuk rangkaian sistem minimum, rangkaian penurun tegangan dan rangkaian USB to TTL yang dibutuhkan, diantaranya: IC ATmega8, IC LM7805, sensor DHT11, RF Modem KYL 200U, sepaket rangkaian *aftermarket* USB to TTL, dan komponen pendukung lainnya, seperti soket, saklar, terminal adaptor 12V, Crystal 16 Mhz, resistor, kapasitor, transistor, dioda, led, dll.
- 2) PCB fiber.
- 3) Mur dan baut.
- 4) Kabel.

- 5) Adaptor 12V.
- 6) Saklar on/off.
- 7) *Box* palstik.

## **E. Langkah Pembuatan Alat**

Langkah pembuatan alat pada proyek akhir ini terdiri dari pelarutan PCB, pemasangan komponen pada PCB, pemasangan rangkaian pada *box*.

### **1. Pembuatan PCB**

#### **a. Pembuatan *layout* PCB**

Langkah awal pembuatan PCB adalah menggambar rangkaian dan *layout* dengan perangkat lunak ISIS dan ARES Profesional. Hasil penggambaran *layout* PCB dapat di lihat pada lampiran.

#### **b. Penyablonan PCB**

Setelah *layout* selesai dibuat maka langkah selanjutnya yaitu menyablonkan *layout* ke PCB polos. Proses penyablonan dilakukan dengan cara :

- 1) Mencetak *layout* pada kertas glossi.
  - 2) Desain *layout* yang sudah dicetak pada kertas glossi disablonkan ke PCB dengan cara disetrika selama kurang lebih 10 menit.
- Setelah gambar *layout* menempel pada PCB, maka hilangkan kertas yang menempel pada PCB dengan air sampai bersih.

#### **c. Pelarutan dan pengeboran PCB**

Langkah selanjutnya yaitu melarutkan PCB dengan larutan *feri chloride* hingga jalur rangkaian terbentuk. Kemudian setelah jalur terbentuk mengangkat PCB dari cairan *Ferri Chloride* dan membersihkannya dengan air. Setelah bersih PCB dibor sesuai

dengan titik-titik yang telah ditentukan.

#### **d. Pemasangan Komponen**

Memasang seluruh komponen yang terdapat pada rangkaian dengan urutan:

1. Menyiapkan komponen yang dibutuhkan.
2. Memasang komponen dari ukuran paling kecil terlebih dahulu.
3. Menyolder kaki komponen sampai semua komponen terpasang.
4. Memotong kaki komponen agar rapi.
5. Menguji rangkaian apakah sudah dapat bekerja dengan baik atau belum.

#### **e. Perancangan Box Alat**

Penempatan alat pada pendeteksi dini kebakaran hutan ini memanfaatkan *box* plastik yang dijual ditoko-toko elektronik. *Box* ini cukup dilubangi untuk menempatkan antena *transceiver* pada alat pemancarnya dan *box* dilubangi kecil pada bagian kanan untuk menempatkan saklarnya, dan dilubangi di bawah saklar untuk menempatkan terminal adaptor sedangkan pada alat penerimanya juga dilubangi untuk menempatkan antenanya dan juga dilubangi persegi panjang kecil untuk menempatkan colok USB, dilubangi kecil-kecil pada bagian sekitar sensor DHT11nya agar sensor dapat membaca suhu dan kelembaban disekitar alat ditempatkan.

**f. Spesifikasi Alat**

*Prototype* telemetri alat pendeteksi dini kebakaran hutan berbasis mikrokontroller ATmega8 mempunyai spesifikasi sebagai berikut :

1. Tegangan kerja VDC.
2. Menggunakan modul KYL, dan sensor suhu dan kelembaban.

**g. Jadwal Pelaksanaan**

Jadwal pelaksanaan pembuatan proyek akhir ini dimulai dari bulan Februari 2015 sampai Oktober 2015 dengan rincian pelaksanaan kegiatan seperti tabel 7:

Tabel 7. Jadwal Pelaksanaan

No.	Kegiatan	Bulan Ke -							
		I	II	II	IV	V	VI	VII	VIII
1	Tahap Perencanaan								
2	Pembuatan <i>Hardware</i>								
3	Pembuatan <i>Software</i>								
4	Pengambilan Data								
5	Penyusunan Laporan								
6	Ujian Proyek Akhir dan Revisi								

#### h. Daftar Komponen yang Dibutuhkan

Berikut ini adalah daftar komponen beserta rincian biaya yang digunakan untuk membuat *Prototype* Telemetri Alat Pendeteksi Dini Kebakaran Hutan Menggunakan ATmega8 Dengan Antarmuka Komputer yang dapat dilihat pada tabel 8:

Tabel 8. Komponen dan biaya proyek akhir

No.	Keterangan	Jumlah	Satuan	Harga	
				Satuan(Rp)	Jumlah(Rp)
1	Modul KYL 200U	2	buah	360000	720000
2	Konektor USB to TTL	1	buah	35000	35000
3	Tombol <i>reset</i>	1	buah	500	500
4	Kabel Pelangi 6 jalur	1	meter	5000	5000
5	IC ATmega8	1	buah	29000	29000
6	IC 7805	1	buah	2000	2000
7	Sensor DHT11	1	buah	40000	40000
8	LED	4	buah	300	1200
9	220uF/16v	1	buah	1200	1200
10	100uF/25v	1	buah	300	300
11	22pF	3	buah	100	300
12	10nF	1	buah	100	100
13	Resistor 330 $\Omega$	1	buah	300	300
14	Resistor 68 $\Omega$	2	buah	300	600
15	Crystal 16MHz	1	buah	2200	2200
16	Transistor C829	2	buah	500	1000
17	Soket IC 28 pin	1	buah	1200	1200
18	Saklar on/off	1	buah	500	500
19	Adaptor	1	buah	35000	35000
20	Box plastik	2	buah	3000	6000
Jumlah					881400

### i. Pengujian Alat

Tabel 9 menjelaskan pengujian kerja dari suplai tegangan. Pengujian ini dilakukan dengan tujuan apakah suplai tegangan yang terukur sudah sesuai dengan tegangan yang seharusnya.

Tabel 9. Pengujian Kerja Suplai Tegangan

No.	Suplai Tegangan	Seharusnya	Terukur	Keterangan
1	Power Supply			
2	Penurun Tegangan			

Pada tabel 9 adalah proses pengukuran dari suplai tegangan yang akan dibahas pada bab IV. Dari pengukuran ini dapat dihitung dengan rumus toleransi kesalahan sebagai berikut:

1. Perhitungan prosentasi kesalahan pada suplai tegangan

$$= \left| \frac{\text{Tegangan Seharusnya} - \text{Tegangan Terukur}}{\text{Tegangan Seharusnya}} \right| \times 100\%$$

2. Perhitungan prosentasi kesalahan pada penurun tegangan

$$= \left| \frac{\text{Tegangan Seharusnya} - \text{Tegangan Terukur}}{\text{Tegangan Seharusnya}} \right| \times 100\%$$

Pengujian alat dilakukan untuk mendapatkan data penelitian. Dalam pengujian alat ini dilakukan dengan dua pengujian, yaitu :

### 1) Uji Fungsional dari *Interface*

Tabel 10. Pengujian Fungsional dari Alat

#### A. Kondisi *Connect*

No.	Pengujian Suhu	Tampilan <i>Interface</i> (terukur)	Keterangan
1	30°		
2	35°		
3	40°		

bersambung...



lanjutan tabel 10...

4	45°		
5	51°		
6	55°		

#### B. Kondisi *Disconnect*

No.	Pengujian Suhu	Tampilan <i>Interface</i> (terukur)	Keterangan
1	30° - 55°		

Tabel 10 menjelaskan pengujian fungsional dari *interface* yang dilakukan dengan cara menguji apa yang terjadi setelah suhu mencapai 50° atau lebih. Pengujian ini dilakukan untuk mengetahui apakah program dari perangkat telah bekerja dengan baik apa belum.

Dilakukan dua kali pengujian alat, pengujian yang pertama yaitu pengujian saat kondisi *connect* dan pengujian yang kedua yaitu pengujian saat kondisi *disconnect*. Hasil pengujian dan pembahasan terdapat pada BAB IV.

#### 2) Uji Keseluruhan Sistem

Pengujian berdasarkan fungsi dan tujuannya dilakukan untuk mengetahui tingkat keberhasilan alat. Hasil dari alat tidak hanya berupa kesuksesan, namun juga ada kekurangan alat. Kekurangan ini yang diharapkan dapat diperbaiki di lain kesempatan. Hasil pengujian dan pembahasan terdapat pada BAB IV.

## BAB IV

### PENGUJIAN DAN PEMBAHASAN

#### A. HASIL PENGUJIAN

Pengujian dilakukan setelah alat yang berjudul “*Prototype* Telemetri Alat Pendeteksi Dini Kebakaran Hutan Menggunakan ATmega8 dengan Antarmuka Komputer” selesai dibuat. Pengujian mempunyai tujuan untuk melihat tingkat keberhasilan dari alat tersebut. Proses pengujian alat ini dilakukan pada setiap bagian atau blok rangkaian dengan menggunakan sebuah multimeter. Pengujian dilakukan pada blok sistem minimum mikrokontroler.

##### 1. Rangkaian Regulator

Rangkaian regulator yang berfungsi untuk menurunkan tegangan dari adaptor 12V menjadi 5V yang sangat diperlukan untuk menjadi sumber tenaga daya keseluruhan rangkaian. Pengukuran dilakukan pada *input* dari adaptor dan *output* pada keluaran dari rangkaian regulator, hal ini bertujuan untuk mengetahui besarnya tegangan yang masuk sebelum ke IC regulator dan tegangan yang keluar dari IC regulator. Hasil pengukuran dapat dilihat pada tabel 11.

Tabel 11. Pengukuran pada IC Regulator

No.	Suplai Tegangan	Seharusnya	Terukur	Keterangan
1	Power Supply	12	11	
2	Penurun Tegangan	5	4.8	

Pada tabel 11 terlihat perbedaan tegangan yang terukur dengan tegangan semestinya. Hal ini dapat terjadi karena komponen yang digunakan memiliki toleransi yang berakibat terjadi perbedaan tegangan. Dari pengukuran ini dapat dihitung toleransi kesalahan pada komponen sebagai berikut:

a. Perhitungan prosentasi kesalahan pada suplai tegangan

$$\begin{aligned}
 &= \left| \frac{\text{Tegangan Seharusnya} - \text{Tegangan Terukur}}{\text{Tegangan Seharusnya}} \right| \times 100\% \\
 &= \left| \frac{12-11}{12} \right| \times 100\% \\
 &= \frac{1}{12} \times 100\% \\
 &= 8,33\%
 \end{aligned}$$

b. Perhitungan prosentasi kesalahan pada penurun tegangan

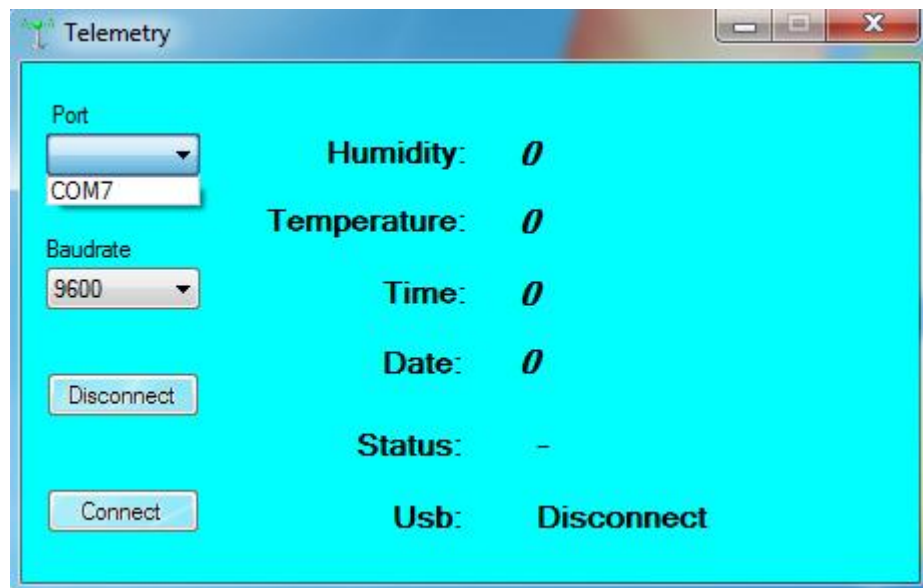
$$\begin{aligned}
 &= \left| \frac{\text{Tegangan Seharusnya} - \text{Tegangan Terukur}}{\text{Tegangan Seharusnya}} \right| \times 100\% \\
 &= \left| \frac{5-4,8}{5} \right| \times 100\% \\
 &= \frac{0,2}{5} \times 100\% \\
 &= 4\%
 \end{aligned}$$

## 2. Rangkaian USB to TTL

Rangkaian *USB to TTL* yang terpasang dengan radio *receiver* (penerima). Dalam pengujian rangkaian *USB to TTL* yaitu dengan cara mengkoneksikan *port* usb ke komputer. Apabila *port* usb to ttl ini bekerja, maka lampu led pada *receiver* akan berkedip.

Setelah itu klik pada “LP\_PORT” yang ada di *desktop*. *Software* ini berguna untuk memberikan perintah untuk menghubungkan dan memutuskan koneksi ke *transmitter* yang dipasang dilain tempat untuk mengukur suhu dan kelembaban. Sebelum mengkoneksikannya, akan

ada perintah untuk memastikan bahwa *port* usb telah terhubung oleh komputer, bisa dilihat pada gambar 25.



Gambar 25. *Interface saat Mengidentifikasi Port*

**c. Rangkaian Sistem Minimum Mikrokontroller ATmega8**

Pengujian sistem minimum ATmega8 dilakukan dengan melihat apakah led indikator pada sistem minimum yang sudah menyala. Selain itu juga dapat dilakukan dengan melihat hasil tampilan LCD apabila sistem minimum sudah dihubungkan dengan LCD. Jika tampilan pada LCD sudah sesuai dengan program maka sistem minimum dikatakan sudah berhasil atau benar. Pada rangkaian sistem minimum ini juga mengalirkan tegangan ke sensor DHT11 dan *transmitter* pada KYL 200U. Data pengukuran tegangan yang dialirkan ke sensor dan *transmitter* tersebut akan dijelaskan pada tabel 12.

Tabel 12. Pengukuran pada sensor dan *transmitter*

No.	Pengukuran	Hasil Pengukuran	
		Saat dihidupkan	Saat bekerja
1.	Sensor DHT11	4.89 Volt	4.89 Volt
2.	Transmitter KYL 200U	4.6 Volt	4.83 Volt

## B. PEMBAHASAN

Dari hasil pengujian dari masing-masing alat ini diperoleh bahwa rangkaian dapat bekerja sesuai dengan yang diharapkan. Pada beberapa bagian sistem terdapat perbedaan hasil pengukuran dari apa yang telah diperoleh dari perhitungan secara teori maupun *datasheet* komponen. Tetapi perbedaan tersebut tidak terlalu menyebabkan terganggunya kinerja alat. Perbedaan hasil tersebut dikarenakan adanya toleransi nilai komponen yang digunakan dan kesalahan pengukuran dari kondisi alat ukur yang kurang optimal. Berikut pembahasan beberapa fungsi rangkaian yang telah diuji :

### 1. Rangkaian Regulator

Rangkaian penurun tegangan berfungsi sebagai penyuplai tegangan ke semua rangkaian, seperti rangkaian sistem minimum mikrokontroler yang mengatur kerja dari sensor DHT11. Apabila tegangan tidak memenuhi kriteria yang dibutuhkan oleh rangkaian maka akan berpengaruh terhadap kinerja rangkaian.

Rangkaian regulator ini bekerja menggunakan tegangan masukan 6-2Vdc. Regulator ini digunakan untuk menurunkan dan mengubah tegangan sumber 6-12Vdc menjadi tegangan 5Vdc.

Data hasil pengukuran yang tercantum dalam tabel 11, hasil pengukuran tegangan masukan dari *power supply* yang seharusnya sebesar 12Vdc menjadi tegangan keluaran sebesar 11Vdc saat diukur. Pengukuran kedua dari penurun tegangan yang seharusnya 5 Volt menjadi 4.8 Volt saat diukur.

Tegangan yang dihasilkan adaptor sudah tepat sesuai dengan teori tetapi hasilnya saat terukur masih dalam batas normal sehingga tidak mempengaruhi kinerja alat. Hal-hal yang mempengaruhi perbedaan nilai tegangan ini diantaranya adalah:

- a. Toleransi komponen
- b. Pengukuran yang kurang tepat
- c. Kondisi alat ukur yang tidak maksimal

Setelah pengujian tiap masing-masing blok selesai, selanjutnya dilakukan perakitan dengan menggabungkan masing-masing blok rangkaian sesuai dengan gambar rangkaian keseluruhan sehingga menjadi sebuah kesatuan yang dapat difungsikan sebagai pendeteksi dini kebakaran hutan.

## 2. Pengujian Fungsional Alat

Tabel 13 menjelaskan pengujian yang dilakukan untuk mengetahui apakah program dari perangkat telah bekerja dengan baik atau belum.

Tabel 13. Pengujian Alat

### A. Kondisi *Connect*

No.	Pengujian Suhu	Tampilan <i>Interface</i> (terukur)	Keterangan
1	30°	 <p>The screenshot shows the 'Telemetry' window with a blue background. On the left, there are dropdown menus for 'Port' (set to COM7) and 'Baudrate' (set to 9600). Below these are 'Disconnect' and 'Connect' buttons. On the right, the following data is displayed: Humidity: 44 %, Temperature: 30 Celcius, Time: 12:25:21 AM, Date: 9/1/2015, Status: Aman, and Usb: Connect.</p>	
2	35°	 <p>The screenshot shows the 'Telemetry' window with a blue background. On the left, there are dropdown menus for 'Port' (set to COM7) and 'Baudrate' (set to 9600). Below these are 'Disconnect' and 'Connect' buttons. On the right, the following data is displayed: Humidity: 41 %, Temperature: 35 Celcius, Time: 12:26:45 AM, Date: 9/1/2015, Status: Aman, and Usb: Connect.</p>	

bersambung...




lanjutan tabel 13...

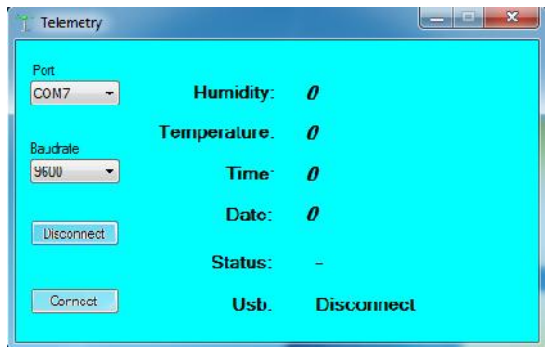
3	40°	 <p>The screenshot shows the Telemetry software interface. On the left, there are dropdown menus for 'Port' (set to COM7) and 'Baudrate' (set to 9600), along with 'Disconnect' and 'Connect' buttons. On the right, the following data is displayed: Humidity: 38 %, Temperature: 40 Celcius, Time: 12:29:33 AM, Date: 9/1/2015, Status: Aman, and Usb: Connect.</p>	
4	45°	 <p>The screenshot shows the Telemetry software interface. On the left, there are dropdown menus for 'Port' (set to COM7) and 'Baudrate' (set to 9600), along with 'Disconnect' and 'Connect' buttons. On the right, the following data is displayed: Humidity: 33 %, Temperature: 45 Celcius, Time: 12:30:40 AM, Date: 9/1/2015, Status: Aman, and Usb: Connect.</p>	
5	51°	 <p>The screenshot shows the Telemetry software interface. On the left, there are dropdown menus for 'Port' (set to COM7) and 'Baudrate' (set to 9600), along with 'Disconnect' and 'Connect' buttons. On the right, the following data is displayed: Humidity: 21 %, Temperature: 51 Celcius, Time: 12:38:02 AM, Date: 9/1/2015, Status: Warning !, and Usb: Connect.</p>	

bersambung...

lanjutan tabel 13...

6	55°		
---	-----	--	--

#### B. Kondisi *Disconnect*

No.	Pengujian Suhu	Tampilan <i>Interface</i> (terukur)	Keterangan
1	30° - 55°		

Tabel 13 menjelaskan kondisi *interface* saat *connect* dan *disconnect*. Saat *connect* terlihat perubahan suhu yang naik sehingga pada suhu diatas 50 akan ada status “Warning !”. Menurut Saharjo (1997), pada pagi hari dengan suhu yang cukup rendah sekitar 20°C ditambah dengan rendahnya kecepatan angin membuat api tidak berkembang sehingga terkonsentrasi pada satu titik. Sementara siang hari dengan suhu 30°-35°C, sedangkan

kadar air bahan bakar cukup rendah (<30%) membuat proses pembakaran berlangsung cepat dan bentuk kebakarannya pun tidak satu titik, tapi berubah-ubah karena pengaruh angin.

### 3. Perangkat Keras (*Hardware*)

*Hardware* dalam sistem ini terdiri dari beberapa bagian yaitu: USB to TTL, rangkaian penurun tegangan, dan sistem minimum.

#### a. Adaptor

Dalam pengukuran catu daya yang bersumber dari adaptor, mengalami penurunan tegangan baik sebelum diberi ataupun telah diberi beban. Penurunan tegangan terjadi akibat dari komponen-komponen yang digunakan dan memiliki nilai toleransi, dan tidak selalu tepat nilai resistansinya. Selain itu juga karena ketelitian dari alat ukur dan pembacaan pada alat ukur.

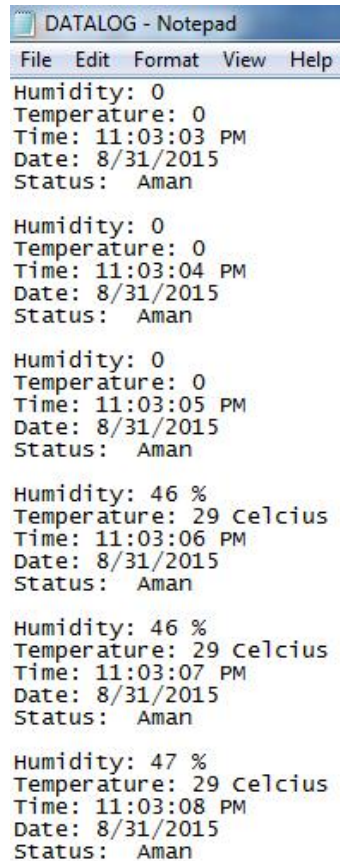
#### b. RF (*Radio Frequency*) Modem

Pengujian pada RF Modem ini dilakukan dengan menghubungkan rangkaian USB to TTL ke komputer. Pada saat dihubungkan, komputer akan mendeteksi *port* dan menginisialisasi *port* sampai benar-benar terhubung. Setelah itu buka *interface* pada komputer/laptop yang digunakan, kemudian koneksikan dengan menekan tombol *connect*. Untuk lebih jelasnya bisa dilihat pada gambar 26.



Gambar 26. *Interface* pada Komputer

Apabila *interface* berjalan dan mencatat data suhu dan kelembaban yang kemudian muncul *notepad* yang bernama "DATALOG" yang tersimpan secara otomatis di *drive* D pada komputer yang berguna untuk menyimpan data suhu dan kelembaban tersebut, maka rangkaian USB to TTL ini dikatakan sudah berhasil atau benar. Untuk lebih jelasnya bisa dilihat pada gambar 27.



```
DATALOG - Notepad
File Edit Format View Help
Humidity: 0
Temperature: 0
Time: 11:03:03 PM
Date: 8/31/2015
Status: Aman

Humidity: 0
Temperature: 0
Time: 11:03:04 PM
Date: 8/31/2015
Status: Aman

Humidity: 0
Temperature: 0
Time: 11:03:05 PM
Date: 8/31/2015
Status: Aman

Humidity: 46 %
Temperature: 29 Celcius
Time: 11:03:06 PM
Date: 8/31/2015
Status: Aman

Humidity: 46 %
Temperature: 29 Celcius
Time: 11:03:07 PM
Date: 8/31/2015
Status: Aman

Humidity: 47 %
Temperature: 29 Celcius
Time: 11:03:08 PM
Date: 8/31/2015
Status: Aman
```

Gambar 27. *Datalog* pada *Interface* saat Merekam Data

Setelah pengujian *software* untuk mengetahui bahwa pada RF Modem tersebut berjalan baik, akan dilakukan pengujian RF Modem dalam proses mengirim dan menerima data. Pengujian ini akan dilakukan pengujian *Line Of Sigh* (LOS) dan *Non-Line Of Sigh* (NLOS) yang merupakan pengujian pengiriman data dengan parameter jarak yang bertujuan untuk mengetahui seberapa jauh data dapat dikirim dari Tx ke Rx tanpa adanya halangan (*obstacle*) dan dengan adanya penghalang (*obstacle*). Setelah pengukuran jarak dilakukan, maka diperoleh data seperti tabel 14 dan tabel 15.

Tabel 14. Jarak koneksi KYL 200U kondisi LOS

No.	Jarak (m)	Status
1	20	<i>Connected</i>
2	40	<i>Connected</i>
3	60	<i>Connected</i>
4	80	<i>Connected</i>
5	100	<i>Connected</i>
6	120	<i>Connected</i>
7	140	<i>Connected</i>
8	160	<i>Connected</i>
9	180	<i>Connected</i>
10	200	<i>Connected</i>
11	210	<i>Connected</i>
12	220	<i>Connected</i>
13	230	<i>Disconnected</i>
14	240	<i>Disconnected</i>
15	250	<i>Disconnected</i>

Tabel 15. Jarak koneksi KYL 200U kondisi NLOS

No.	Jarak (m)	Status
1	20	<i>Connected</i>
2	40	<i>Connected</i>
3	60	<i>Connected</i>
4	80	<i>Connected</i>
5	100	<i>Connected</i>
6	120	<i>Connected</i>
7	140	<i>Connected</i>
8	160	<i>Connected</i>
9	180	<i>Connected</i>
10	200	<i>Disconnected</i>
11	210	<i>Disconnected</i>
12	220	<i>Disconnected</i>
13	230	<i>Disconnected</i>
14	240	<i>Disconnected</i>
15	250	<i>Disconnected</i>

Pada tabel 14 dan tabel 15 dapat dianalisa bahwa saat kondisi LOS, jarak terjauh antara Tx ke Rx dalam transmisi data sejauh 220m. Lebih dari 220m, maka data tidak akan diterima oleh Rx. Pada

saat kondisi NLOS, jarak terjauh antara Tx ke Rx dalam transmisi data sejauh 180m. Lebih dari 180m, maka data tidak akan diterima oleh Rx

c. Sistem Minimum

Pengujian sistem minimum ATmega8 dilakukan dengan melihat apakah led indikator berwarna biru yang terpasang pada rangkaian sistem minimum sudah menyala. Selain itu juga dapat dilakukan dengan melihat hasil tampilan di komputer apabila sistem minimum sudah dihubungkan dengan komputer. Jika tampilan pada komputer sudah sesuai dengan program maka sistem minimum dikatakan sudah berhasil atau benar. Lebih jelasnya dapat dilihat pada gambar 28.



Gambar 28. Pengujian pada Sistem Minimum ATmega8

d. Rangkaian Sensor

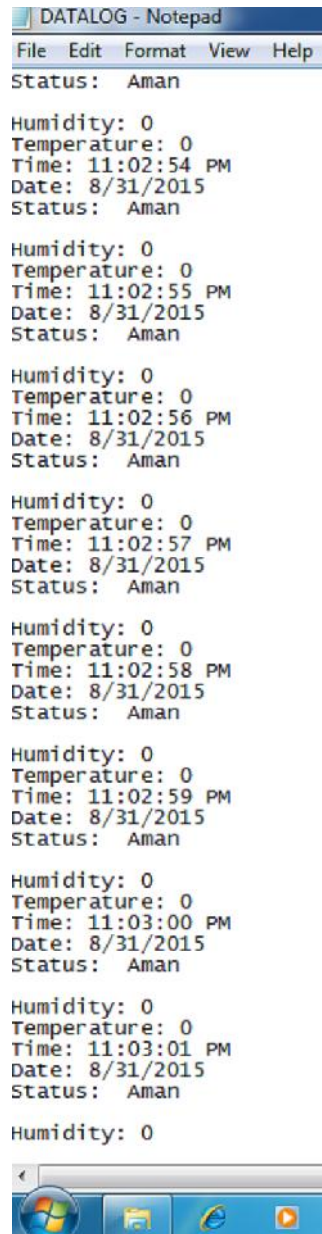
Berdasarkan hasil pengujian Sensor bekerja mendeteksi suhu dan kelembaban ketika ada perintah dari komputer untuk



menjalankan alat. Saat mendapat perintah *connect*, pada keadaan akan bekerja sensor akan *loading* selama  $\pm 10$  detik untuk membaca program kemudian bekerja dan langsung mengirimkan data suhu dan kelembaban yang terukur ke komputer. Saat saklar pada posisi *on*, lampu led smd pada sensor DHT11 akan menyala merah, lampu indikator led biru akan menyala dan lampu led hijau pada rangkaian mikrokontroller akan berkedip saat alat bekerja. Lebih jelasnya dapat dilihat pada gambar 29 dan gambar 30.



Gambar 29. Pengujian saat Sensor DHT11 Bekerja



Gambar 30. Proses *Loading* Sensor Selama 10 Detik

#### 4. Perangkat Lunak (*Software*)

Perangkat lunak atau program merupakan bagian penting karena tanpa adanya perangkat lunak alat pada proyek akhir ini tidak akan bisa bekerja. Pembuatan perangkat lunak menggunakan *software* Microsoft Visual Studio 2010 (Visual Basic) pada *interface* untuk memantau dari

komputer, sedangkan pada mikrokontrollernya menggunakan Arduino Compiler. Konsep utama dari program yang dibuat adalah pembacaan sensor, kirim data dan terima data. Berikut akan dijelaskan bagian-bagian dari program yang dibuat.

a. Program pada *interface*

```
Imports System
Imports System.IO.Ports
Imports System.Threading
Imports System.ComponentModel
```

Baris instruksi diatas menjelaskan *input library system*, *input sistem pada IO Port (port input output)*, *input system threading* dan *input sistem pada model komponen*. Proses ini digunakan untuk pengambilan masukan dari setiap sistem yang akan dikerjakan.

```
Public Class Form1
    Dim kelembaban As Double
    Dim suhu As Double
    Dim myport As Array
```

Baris instruksi diatas merupakan proses pengaturan *port variable*. *Port variable* yang dimaksud akan dijelaskan pada bagian instruksi selanjutnya.

```

Private Sub Form1_Load (ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load

    myport = IO.Ports.SerialPort.Get Port Names ()
    port.Items.Add Range (myport)

    port.Text = "COM1"
    baudrate.Text = "9600"
    keamanan.Text = " - "
    koneksi.Text = " Disconnect "

    If Close Reason.Application Exit Call Then

        rx_tx.Close ()

    End If

End Sub

```

Baris instruksi diatas merupakan bagian program yang disebut dengan *load* yang digunakan untuk memuat *port serial* yang menggunakan *default port* COM1 dan mengisi pengaturan secara otomatis pada *baudrate* yang digunakan yaitu sebesar 9600.

```

Private Sub connect_Click (ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles connect.Click

    disconnect.BackColor = Color.Gold

    koneksi.Text = " Connect "

    Timer1.Enabled = True

    rx_tx.Port Name = port.Text

    rx_tx.BaudRate = baudrate.Text

    rx_tx.Open ()

    port.Enabled = False

    baudrate.Enabled = False

    connect.Enabled = False

End Sub

```

Baris instruksi diatas merupakan bagian program yang difungsikan saat tombol “*connect*” pada *interface* di klik. Program ini menjelaskan langkah awal yang harus dilakukan untuk mengkoneksikan perangkat yaitu pemilihan *port* COM dan pemilihan *baudrate*. Arti “*False*” pada “*port.Enabled*”, “*baudrate.Enabled*” dan “*connect.Enabled*” ini ialah supaya saat tombol *connect* di klik pada pemilihan *port*, pemilihan *baudrate* dan tombol *connect* tidak bisa di ubah melainkan hanya sekali klik saja. Saat posisi “*connect*” akan ada perubahan warna pada tombol “*disconnect*” yaitu berubah menjadi warna emas. Perintah ini ada pada baris program “*disconnect.BackColor = Color.Gold*”.

```

Private Sub disconnect_Click (ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles disconnect.Click

    disconnect.BackColor = Color.White

    koneksi.Text = " Disconnect "

    keamanan.Text = " - "

    Timer1.Enabled = False

    waktu.Text = 0

    tanggal.Text = 0

    humidity.Text = 0

    temperature.Text = 0

    port.Enabled = True

    baudrate.Enabled = True

    connect.Enabled = True

    rx_tx.Close()

End Sub

```

Baris instruksi diatas merupakan bagian program pada saat posisi *"disconnect"*. Program ini menjelaskan keadaan *interface* saat tombol *"disconnect"* pada *interface* di klik. Saat posisi *"disconnect"* akan terjadi perubahan warna *background*, *background* yang semula saat posisi *"connect"* berwarna emas akan berubah menjadi warna putih saat tombol *"disconnect"* di klik. Arti *"True"* pada *"port.Enabled"*, *"baudrate.Enabled"* dan *"connect.Enabled"* ini ialah supaya saat posisi *"disconnect"* di klik maka pemilihan *port*, pemilihan *baudrate* dan tombol *connect* dapat di ubah/dapat di atur kembali.

```

Private Sub rx_tx_Data Received (ByVal sender As
System.Object, ByVal e As System.IO.Ports.Serial Data Received
EventArgs) Handles rx_tx.Data Received

    Dim s As String = rx_tx.Read Line

    If s.Contains ("Hum") Then
        s = s.Replace ("Hum:", "")
        If Is Numeric (s) Then
            kelembaban = CDb1 (s)
            humidity.Invoke (New baca (AddressOf updatebaca),
kelembaban.To String + " %")
        End If

    Else If s.Contains ("Temp") Then
        s = s.Replace ("Temp:", "")
        If Is Numeric (s) Then
            suhu = CDb1 (s)
            temperature.Invoke (New baca1 (AddressOf
updatebaca1), suhu.To String + " Celcius")
        End If

    End If

End Sub

```

Baris instruksi diatas merupakan baris program pada saat data diterima. Saat data kelembaban diterima pada *datalog* tertulis dengan nilai angka yang dinyatakan dalam “%”, sedangkan pada saat data suhu diterima pada *datalog* tertulis dengan nilai angka yang dinyatakan dalam “Celcius”.

```
Delegate Sub baca (ByVal t As String)
    Public Sub updatebaca (ByVal t As String)
        humidity.Text = t
    End Sub

    Delegate Sub baca1 (ByVal t As String)
    Public Sub updatebaca1 (ByVal t As String)
        temperature.Text = t
    End Sub
```

Baris instruksi diatas merupakan baris program pada saat data yang diterima yaitu suhu dan kelembaban akan disimpan secara otomatis. Penyimpanan akan tertulis sesuai urutan program dari yang pertama tertuliskan "humidity.Text = t" akan dituliskan *humidity* beserta data angka yang dinyatakan dalam % dan yang kedua tertuliskan pada program "temperature.Text = t" akan dituliskan *temperature* beserta data angka yang dinyatakan dalam Celcius. Proses ini akan terus berulang selama alat masih dijalankan.



```

Private Sub Timer1_Tick (ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Timer1.Tick

    Dim file As System.IO.StreamWriter

    waktu.Text = TimeOfDay
    tanggal.Text = Date.Today

    If suhu >= 50 Then

        keamanan.Text = " Warning ! "

    else

        keamanan.Text = " Aman "

    End if

    file = My.Computer.FileSystem. _
    OpenTextFileWriter("D:\DATALOG.txt", True)
    file.WriteLine("Humidity: " + humidity.Text, " % ")
    file.WriteLine("Temperature: " + temperature.Text, " Celcius ")
    file.WriteLine("Time: " + waktu.Text)
    file.WriteLine("Date: " + tanggal.Text)
    file.WriteLine("Status: " + keamanan.Text)
    file.WriteLine(" ")
    file.Close()

End Sub

End Class

```

Baris instruksi diatas merupakan baris program yang menginstruksikan pada saat proses penulisan data *humidity*, *temperature*, *time*, *date*, status dan usb tertulis dan tersimpan secara otomatis ke *datalog* di D: pada komputer. Pada saat instruksi penulisan waktu bisa terlihat pada baris program "waktu.Text = TimeOfDay", saat penulisan tanggal pada hari dimana alat itu dijalankan bisa terlihat pada baris program "tanggal.Text = Date.Today", sedangkan pada instruksi program "If suhu >= 50 Then keamanan.Text = " Warning ! " else keamanan.Text = " Aman "End if" ini menginstruksikan pada saat suhu lebih dari atau sama dengan 50 akan ditampilkan status "Warning !" sedangkan jika suhu dibawah 50 akan ditampilkan status "Aman" dan akhirnya semua data yang telah terekam akan disimpan otomatis ke D: di komputer yaitu terlihat pada baris program "OpenTextFileWriter("D:\DATALOG.txt", True)".

b. Program pada mikrokontroller ATmega8

```
#include <dht11.h>

dht11 DHT11 = dht11(D2, BUSA); //we are placing our devide on
D1 on BUSA
```

Baris instruksi diatas merupakan baris program yang menginstruksikan penulisan program yang ada di library fungsi pada DHT11. Pada penulisan program "dht11(D2, BUSA)" ini bermaksud pin khusus untuk menghubungkan dari sensor ke ATmega ada di pin D2.

```
void setup() {  
    Serial.begin(9600);  
    pinMode(13,OUTPUT);  
    digitalWrite(13,HIGH);  
    delay(2000);  
}
```

Baris instruksi diatas merupakan baris program yang menginstruksikan keluaran dari sensor DHT11. Penulisan “void setup() { “ ini sudah di sesuaikan oleh *database* yang ada di sensor dht11, sehingga semua penulisan baris program diatas sudah sesuai oleh *database*. Penulisan baris program “pinMode(13,OUTPUT)” ini maksudnya output di pin 13, “digitalWrite(13,HIGH)” maksudnya untuk menyalakan lampu indikator dan “delay(2000)” adalah jeda pada waktu lampu menyala yaitu selama 2 detik.

```
void loop() {  
    DHT11.read();  
    Serial.print("Hum:");  
    Serial.println((int)DHT11.humidity);  
    Serial.print("Temp:");  
    Serial.println((int)DHT11.temperature);  
    digitalWrite(13,LOW);  
    delay(1000);  
    digitalWrite(13,HIGH);  
    delay(1000);  
}
```

Baris instruksi diatas merupakan baris program yang menginstruksikan proses pembacaan pada sensor DHT11. Baris program ini dapat dilihat pada “DHT11.read();” dan pada penulisan baris program serial.print ini adalah pembacaan data sekaligus penulisan data yang dideteksi oleh sensor DHT11. Penulisan baris program “digitalWrite(13,LOW); dan digitalWrite(13,HIGH);” adalah proses pada saat lampu indikator menyala dan “delay(1000);” adalah proses jeda saat lampu menyala yaitu selama 1 detik.

### **C. UNJUK KERJA**

Alat pendeteksi dini kebakaran hutan dengan antarmuka komputer telah berfungsi dengan baik dikarenakan kinerja di setiap blok rangkaian telah berfungsi secara optimal. Pendeteksi dini kebakaran hutan ini dapat

mendeteksi bila ada perubahan suhu dan kelembaban yang drastis dari hutan tersebut.

Sistem pendeteksi dini kebakaran hutan memiliki 2 bagian pada *hardware*, pada bagian pertama yaitu penerima yang dihubungkan ke komputer melalui *port* usb sedangkan bagian *hardware* satunya adalah pengirim yang memiliki 1 saklar *ON/OFF*, 1 sensor DHT11, dan 1 terminal untuk adaptor 12V. Pendeteksi dini kebakaran hutan diprogram hanya untuk mendeteksi perubahan suhu dan kelembaban.

Alat pendeteksi dini kebakaran hutan akan bekerja dengan menghidupkan saklar dan mengkoneksikannya dengan menekan tombol *connect* pada *interface* yang sudah terinstall di komputer. Pada saat dihidupkan, sensor akan mengalami *loading* selama 10 detik, kemudian apabila alat sedang bekerja indikator led hijau akan menyala berkedip sedangkan saat proses mengirim dan menerima data indikator led merah pada *receiver* akan berkedip.

## BAB V

### KESIMPULAN DAN SARAN

#### A. KESIMPULAN

Bedasarkan dari uraian perancangan, pembuatan dan pembahasan dari pendeteksi dini kebakaran hutan, maka dapat disimpulkan :

1. Perangkat keras pendeteksi dini kebakaran hutan telah terealisasi dan terdiri dari beberapa rangkaian *input*, unit rangkaian proses dan rangkaian *output*. Unit rangkaian input terdiri dari rangkaian sistem minimum, sensor DHT11, saklar ON/OFF, *transmitter*, dan terminal untuk adaptor 12V. Saklar untuk menghidupkan atau mematikan alat, adaptor untuk sumber daya alat, sensor DHT11 sebagai pendeteksi suhu dan kelembaban, *transmitter* sebagai pengirim. Rangkaian proses terdiri dari sistem minimum ATmega8.
2. Penyusunan perangkat lunak (*software*) pada *prototype* pendeteksi dini kebakaran hutan berbasis mikrokontroler ATmega8 berupa program Visual Basic yang terdiri dari beberapa bagian : definisi prosesor, definisi waktu, definisi *port*, definisi variabel. Sedangkan bahasa C dibuat menggunakan Arduino Compiler yang berfungsi untuk program berhenti dan program utama. Perangkat lunak yang dibuat telah berjalan dengan baik sebagaimana fungsinya.
3. Unjuk kerja *prototype* telemetri alat pendeteksi dini kebakaran hutan secara keseluruhan sudah bekerja sesuai harapan. Hal ini dibuktikan pada pengamatan dan pengujian pada bab IV. Persentase kesalahan yang terjadi hanya sebesar 4%.

## B. KETERBATASAN ALAT

Alat yang telah dibuat ini masih memiliki keterbatasan, yaitu :

1. Pada RF Modem KYL 200U yang digunakan hanya dapat mendeteksi sensor sejauh 200m saja.
2. Pada sensor suhu dan kelembaban yang menggunakan sensor DHT 11 memiliki kelemahan pada *start* awal alat digunakan, pada sensor harus *loading* terlebih dahulu selama  $\pm 10$  detik.
3. Pada *interface* belum canggih karena kadang *software* saat proses dihentikan sering terjadi eror.

## C. SARAN

Dalam pembuatan proyek akhir ini terdapat kekurangan seperti yang disebutkan dalam *point* keterbatasan alat, sehingga diperlukan pengembangan guna menyempurnakan proyek akhir ini. Penulis mempunyai beberapa saran untuk menyempurnakan alat ini dari keterbatasan diantaranya :

1. Mengubah rancangan mekanik yang dapat mendeteksi perubahan lainnya yang hanya tidak mendeteksi perubahan suhu dan kelembaban saja.
2. Memilih sensor suhu dan kelembaban yang baik tanpa harus *loading* dahulu saat akan digunakan.
3. Pemilihan *transceiver* yang bisa digunakan untuk jarak jauh.

## DAFTAR PUSTAKA

Djuandi Feri. (2011). Pengenalan Arduino. Jakarta: P.T. Elexmedia Komputindo.

Janata J. (2009). Principles Of Sensor, 2Ed. Inggris: Springer.

Kanginan Marthen. (2004). Sains Fisika 2 A untuk Kelas VIII. Jakarta: Erlangga.

Lakitan Benyamin. (2002). Dasar-Dasar Klimatologi, cetakan kedua. Jakarta: Grafindo Persada.

P.H. Simale. (1995). *Sistem Telekomunikasi*, Edisi kedua. Jakarta: Erlangga.

S Wasito. (2001). Vademekum Elektronika. Jakarta: P.T. Gramedia Pustaka Utama.

Suratmo. (2003). Pengetahuan Dasar Pengendalian Kebakaran Hutan. Bogor: IPB Press.

Setiawan, Muhammad. (2008). Prototipe Sistem Pendeteksi Kebakaran Hutan Menggunakan Modem Wavecom sebagai pengirim SMS Berbasis Mikrokontroler ATmega16. Universitas Negeri Yogyakarta.

Winoto Ardi. (2008). Mikrokontroller AVR Atmega8/32/16/8535 & Pemrogramannya dengan Bahasa C pada WIN AVR. Jakarta.

Yuswanto & Subari. (2007). Visual Basic .Net 2010. Jakarta.

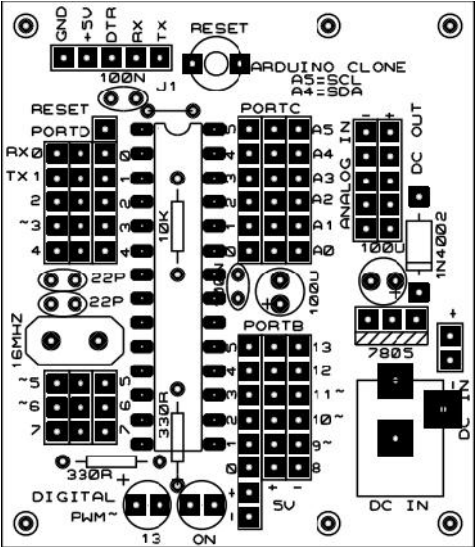
Saharjo,, B.H. (1997). Mengapa Hutan dan Lahan Terbakar. Harian Republika.



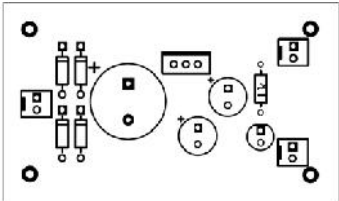
# LAMPIRAN



Lampiran 2. *Layout* Komponen



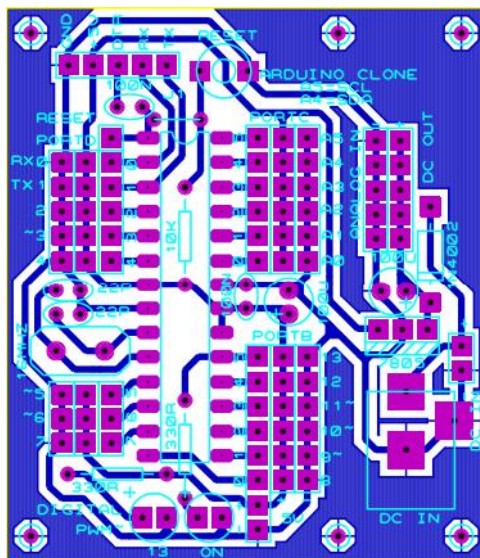
*Layout* Komponen Sismin ATmega8



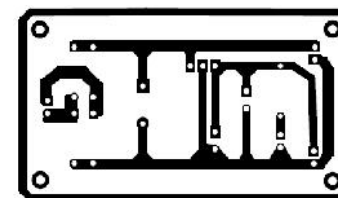
*Layout* Komponen Catu Daya

LAYOUT KOMPONEN			KETERANGAN	
FT UNY	Skala : -	Dig. Enggar	A4	No.2
	Dip. Dessy I.	Dist. Dessy I.	10507131027	

Lampiran 3. *Layout PCB*



*Layout PCB Sistem Minimum ATmega8*



*Layout PCB Catu Daya*

LAYOUT PCB			KETERANGAN	
FT UNY	Skala : -	Dig. Enggar	A4	No.3
	Dip. Dessy I.	Dist. Dessy I.	10507131027	

#### Lampiran 4. Program

##### ***Coding Software***

Imports System

Imports System.IO.Ports

Imports System.Threading

Imports System.ComponentModel

Public Class Form1

Dim kelembaban As Double

Dim suhu As Double

Dim myport As Array

Private Sub Form1\_Load (ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles MyBase.Load

myport = IO.Ports.SerialPort.Get Port Names ()

port.Items.Add Range (myport)

port.Text = "COM1"

baudrate.Text = "9600"

keamanan.Text = " - "

```
koneksi.Text = " Disconnect "
```

```
If Close Reason.Application Exit Call Then
```

```
rx_tx.Close ()
```

```
End If
```

```
End Sub
```

```
Private Sub connect_Click (ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles connect.Click
```

```
disconnect.Back Color = Color.Gold
```

```
koneksi.Text = " Connect "
```

```
Timer1.Enabled = True
```

```
rx_tx.Port Name = port.Text
```

```
rx_tx.BaudRate = baudrate.Text
```

```
rx_tx.Open ()
```

```
port.Enabled = False
```

```
baudrate.Enabled = False
```

```
connect.Enabled = False
```

```
End Sub
```

```
Private Sub disconnect_Click (ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles disconnect.Click
```

```

disconnect.BackColor = Color.White
koneksi.Text = " Disconnect "
keamanan.Text = " - "
Timer1.Enabled = False
waktu.Text = 0
tanggal.Text = 0
humidity.Text = 0
temperature.Text = 0
port.Enabled = True
baudrate.Enabled = True
connect.Enabled = True
rx_tx.Close()

```

```

End Sub

```

```

Private Sub rx_tx_Data Received (ByVal sender As System.Object, ByVal e
As System.IO.Ports.Serial Data Received EventArgs) Handles rx_tx.Data
Received

```

```

    Dim s As String = rx_tx.Read Line

```

```

    If s.Contains ("Hum") Then

```

```

        s = s.Replace ("Hum:", "")

```

```

        If Is Numeric (s) Then

```

```

        kelembaban = CDbI (s)

        humidity.Invoke (New baca (AddressOf updatebaca), kelembaban.To
String + " %")

    End If

```

```

Else If s.Contains ("Temp") Then

    s = s.Replace ("Temp:", "")

    If Is Numeric (s) Then

        suhu = CDbI (s)

        temperature.Invoke (New baca1 (AddressOf updatebaca1), suhu.To
String + " Celcius")

    End If

```

```

End If

```

```

End Sub

```

```

Delegate Sub baca (ByVal t As String)

Public Sub updatebaca (ByVal t As String)

    humidity.Text = t

End Sub

```

```

Delegate Sub baca1 (ByVal t As String)

Public Sub updatebaca1 (ByVal t As String)

    temperature.Text = t

```



End Sub

Private Sub Timer1\_Tick (ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Timer1.Tick

Dim file As System.IO.StreamWriter

waktu.Text = TimeOfDay

tanggal.Text = Date.Today

If suhu >= 50 Then

keamanan.Text = " Warning ! "

else

keamanan.Text = " Aman "

End if

file = My.Computer.FileSystem. \_

OpenTextFileWriter("D:\DATALOG.txt", True)

file.WriteLine("Humidity: " + humidity.Text, " % ")

file.WriteLine("Temperature: " + temperature.Text, " Celcius ")

file.WriteLine("Time: " + waktu.Text)

file.WriteLine("Date: " + tanggal.Text)

file.WriteLine("Status: " + keamanan.Text)

file.WriteLine(" ")

file.Close()

End Sub

End Class

## **Coding ATmega**

```
#include <dht11.h>
```

```
dht11 DHT11 = dht11(D2, BUSA); //we are placing our device on D1 on BUSA
```

```
void setup() {  
    Serial.begin(9600);  
    pinMode(13, OUTPUT);  
    digitalWrite(13, HIGH);  
    delay(2000);  
}  
void loop() {  
    DHT11.read();  
    Serial.print("Hum:");  
    Serial.println((int)DHT11.humidity);  
    Serial.print("Temp:");  
    Serial.println((int)DHT11.temperature);  
    digitalWrite(13, LOW);  
    delay(1000);  
    digitalWrite(13, HIGH);  
    delay(1000);  
}
```

## Features

- High-performance, Low-power Atmel® AVR® 8-bit Microcontroller
- Advanced RISC Architecture
  - 130 Powerful Instructions – Most Single-clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 16MIPS Throughput at 16MHz
  - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory segments
  - 8Kbytes of In-System Self-programmable Flash program memory
  - 512Bytes EEPROM
  - 1Kbyte Internal SRAM
  - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - Data retention: 20 years at 85°C/100 years at 25°C<sup>(1)</sup>
  - Optional Boot Code Section with Independent Lock Bits
  - In-System Programming by On-chip Boot Program
  - True Read-While-Write Operation
  - Programming Lock for Software Security
- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescaler, one Compare Mode
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Three PWM Channels
  - 8-channel ADC in TQFP and QFN/MLF package
    - Eight Channels 10-bit Accuracy
  - 6-channel ADC in PDIP package
    - Six Channels 10-bit Accuracy
  - Byte-oriented Two-wire Serial Interface
  - Programmable Serial USART
  - Master/Slave SPI Serial Interface
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated RC Oscillator
  - External and Internal Interrupt Sources
  - Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby
- I/O and Packages
  - 23 Programmable I/O Lines
  - 28-lead PDIP, 32-lead TQFP, and 32-pad QFN/MLF
- Operating Voltages
  - 2.7V - 5.5V (ATmega8L)
  - 4.5V - 5.5V (ATmega8)
- Speed Grades
  - 0 - 8MHz (ATmega8L)
  - 0 - 16MHz (ATmega8)
- Power Consumption at 4MHz, 3V, 25°C
  - Active: 3.6mA
  - Idle Mode: 1.0mA
  - Power-down Mode: 0.5µA



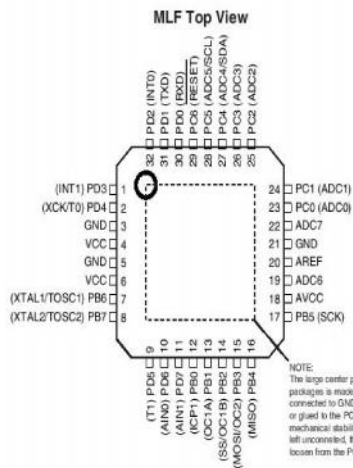
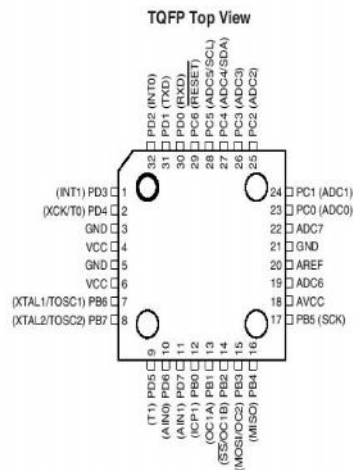
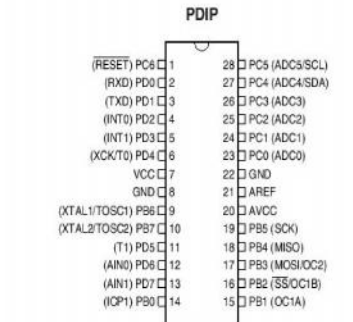
**8-bit Atmel with  
8KBytes In-  
System  
Programmable  
Flash**

**ATmega8  
ATmega8L**

Rev. 2486AA-AVR-02/2013



## Pin Configurations

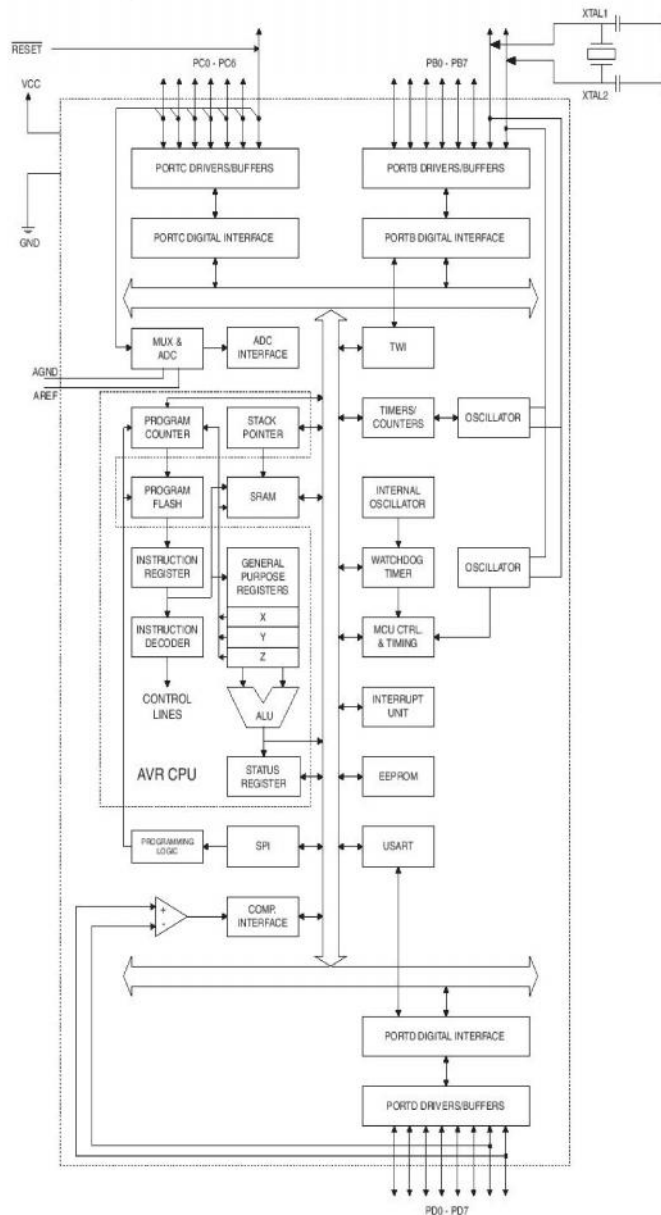


## Overview

The Atmel®AVR® ATmega8 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8 achieves throughputs approaching 1MIPS per MHz, allowing the system designer to optimize power consumption versus processing speed.

## Block Diagram

Figure 1. Block Diagram



The Atmel®AVR® core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega8 provides the following features: 8 Kbytes of In-System Programmable Flash with Read-While-Write capabilities, 512 bytes of EEPROM, 1 Kbyte of SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte oriented Two-wire Serial Interface, a 6-channel ADC (eight channels in TQFP and QFN/MLF packages) with 10-bit accuracy, a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next Interrupt or Hardware Reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption.

The device is manufactured using Atmel's high density non-volatile memory technology. The Flash Program memory can be reprogrammed In-System through an SPI serial interface, by a conventional non-volatile memory programmer, or by an On-chip boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash Section will continue to run while the Application Flash Section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega8 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications.

The ATmega8 is supported with a full suite of program and system development tools, including C compilers, macro assemblers, program simulators, and evaluation kits.

#### **Disclaimer**

Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Minimum and Maximum values will be available after the device is characterized.

## Pin Descriptions

**VCC** Digital supply voltage.

**GND** Ground.

**Port B (PB7..PB0)**  
**XTAL1/XTAL2/TOSC1/TOSC2**

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB7..6 is used as TOSC2..1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

The various special features of Port B are elaborated in [“Alternate Functions of Port B” on page 58](#) and [“System Clock and Clock Options” on page 25](#).

**Port C (PC5..PC0)**

Port C is an 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

**PC6/RESET**

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. The minimum pulse length is given in [Table 15 on page 38](#). Shorter pulses are not guaranteed to generate a Reset.

The various special features of Port C are elaborated on [page 61](#).

**Port D (PD7..PD0)**

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Port D also serves the functions of various special features of the ATmega8 as listed on [page 63](#).

**RESET**

Reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length is given in [Table 15 on page 38](#). Shorter pulses are not guaranteed to generate a reset.



<b>AV<sub>CC</sub></b>	AV <sub>CC</sub> is the supply voltage pin for the A/D Converter, Port C (3..0), and ADC (7..6). It should be externally connected to V <sub>CC</sub> , even if the ADC is not used. If the ADC is used, it should be connected to V <sub>CC</sub> through a low-pass filter. Note that Port C (5..4) use digital supply voltage, V <sub>CC</sub> .
<b>AREF</b>	AREF is the analog reference pin for the A/D Converter.
<b>ADC7..6 (TQFP and QFN/MLF Package Only)</b>	In the TQFP and QFN/MLF package, ADC7..6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.



**Resources**

A comprehensive set of development tools, application notes and datasheets are available for download on <http://www.atmel.com/avr>.

**Data Retention**

Reliability Qualification results show that the projected data retention failure rate is much less than 1 PPM over 20 years at 85°C or 100 years at 25°C.

**About Code Examples**

This datasheet contains simple code examples that briefly show how to use various parts of the device. These code examples assume that the part specific header file is included before compilation. Be aware that not all C compiler vendors include bit definitions in the header files and interrupt handling in C is compiler dependent. Please confirm with the C compiler documentation for more details.

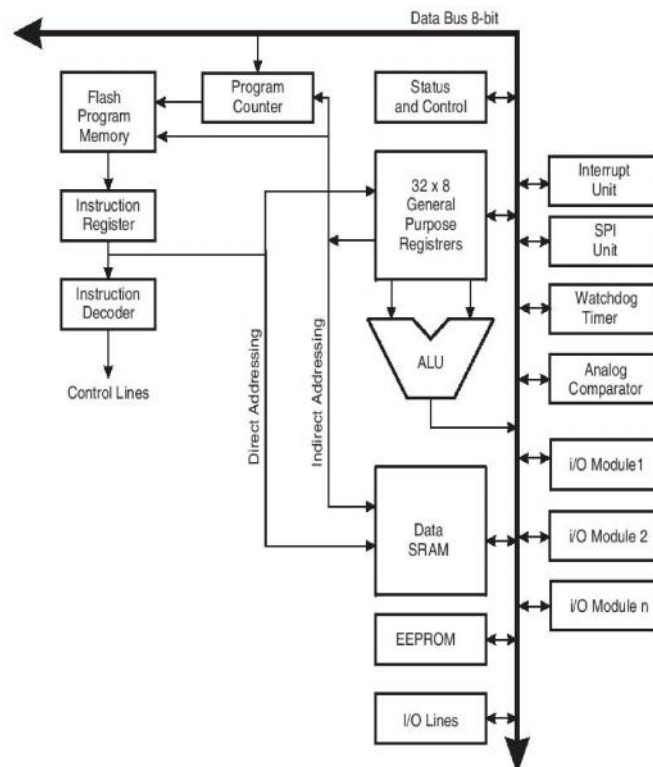
## Atmel AVR CPU Core

### Introduction

This section discusses the Atmel® AVR® core architecture in general. The main function of the CPU core is to ensure correct program execution. The CPU must therefore be able to access memories, perform calculations, control peripherals, and handle interrupts.

### Architectural Overview

Figure 2. Block Diagram of the AVR MCU Architecture



In order to maximize performance and parallelism, the AVR uses a Harvard architecture – with separate memories and buses for program and data. Instructions in the Program memory are executed with a single level pipelining. While one instruction is being executed, the next instruction is pre-fetched from the Program memory. This concept enables instructions to be executed in every clock cycle. The Program memory is In-System Reprogrammable Flash memory.

The fast-access Register File contains 32 x 8-bit general purpose working registers with a single clock cycle access time. This allows single-cycle Arithmetic Logic Unit (ALU) operation. In a typical ALU operation, two operands are output from the Register File, the operation is executed, and the result is stored back in the Register File – in one clock cycle.

Six of the 32 registers can be used as three 16-bit indirect address register pointers for Data Space addressing – enabling efficient address calculations. One of these address pointers

can also be used as an address pointer for look up tables in Flash Program memory. These added function registers are the 16-bit X-register, Y-register, and Z-register, described later in this section.

The ALU supports arithmetic and logic operations between registers or between a constant and a register. Single register operations can also be executed in the ALU. After an arithmetic operation, the Status Register is updated to reflect information about the result of the operation.

The Program flow is provided by conditional and unconditional jump and call instructions, able to directly address the whole address space. Most AVR instructions have a single 16-bit word format. Every Program memory address contains a 16-bit or 32-bit instruction.

Program Flash memory space is divided in two sections, the Boot program section and the Application program section. Both sections have dedicated Lock Bits for write and read/write protection. The SPM instruction that writes into the Application Flash memory section must reside in the Boot program section.

During interrupts and subroutine calls, the return address Program Counter (PC) is stored on the Stack. The Stack is effectively allocated in the general data SRAM, and consequently the Stack size is only limited by the total SRAM size and the usage of the SRAM. All user programs must initialize the SP in the reset routine (before subroutines or interrupts are executed). The Stack Pointer SP is read/write accessible in the I/O space. The data SRAM can easily be accessed through the five different addressing modes supported in the AVR architecture.

The memory spaces in the AVR architecture are all linear and regular memory maps.

A flexible interrupt module has its control registers in the I/O space with an additional global interrupt enable bit in the Status Register. All interrupts have a separate Interrupt Vector in the Interrupt Vector table. The interrupts have priority in accordance with their Interrupt Vector position. The lower the Interrupt Vector address, the higher the priority.

The I/O memory space contains 64 addresses for CPU peripheral functions as Control Registers, SPI, and other I/O functions. The I/O Memory can be accessed directly, or as the Data Space locations following those of the Register File, 0x20 - 0x5F.

## Arithmetic Logic Unit – ALU

The high-performance Atmel® AVR® ALU operates in direct connection with all the 32 general purpose working registers. Within a single clock cycle, arithmetic operations between general purpose registers or between a register and an immediate are executed. The ALU operations are divided into three main categories – arithmetic, logical, and bit-functions. Some implementations of the architecture also provide a powerful multiplier supporting both signed/unsigned multiplication and fractional format. For a detailed description, see ["Instruction Set Summary" on page 311](#).

## Status Register

The Status Register contains information about the result of the most recently executed arithmetic instruction. This information can be used for altering program flow in order to perform conditional operations. Note that the Status Register is updated after all ALU operations, as specified in the [Instruction Set Reference](#). This will in many cases remove the need for using the dedicated compare instructions, resulting in faster and more compact code.

The Status Register is not automatically stored when entering an interrupt routine and restored when returning from an interrupt. This must be handled by software.

The AVR Status Register – SREG – is defined as:

Bit	7	6	5	4	3	2	1	0	
	I	T	H	S	V	N	Z	C	SREG
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

### • Bit 7 – I: Global Interrupt Enable

The Global Interrupt Enable bit must be set for the interrupts to be enabled. The individual interrupt enable control is then performed in separate control registers. If the Global Interrupt Enable Register is cleared, none of the interrupts are enabled independent of the individual interrupt enable settings. The I-bit is cleared by hardware after an interrupt has occurred, and is set by the RETI instruction to enable subsequent interrupts. The I-bit can also be set and cleared by the application with the SEI and CLI instructions, as described in the [Instruction Set Reference](#).

### • Bit 6 – T: Bit Copy Storage

The Bit Copy instructions BLD (Bit Load) and BST (Bit Store) use the T-bit as source or destination for the operated bit. A bit from a register in the Register File can be copied into T by the BST instruction, and a bit in T can be copied into a bit in a register in the Register File by the BLD instruction.

### • Bit 5 – H: Half Carry Flag

The Half Carry Flag H indicates a Half Carry in some arithmetic operations. Half Carry is useful in BCD arithmetic. See the ["Instruction Set Description"](#) for detailed information.

### • Bit 4 – S: Sign Bit, $S = N \oplus V$

The S-bit is always an exclusive or between the Negative Flag N and the Two's Complement Overflow Flag V. See the ["Instruction Set Description"](#) for detailed information.

### • Bit 3 – V: Two's Complement Overflow Flag

The Two's Complement Overflow Flag V supports two's complement arithmetics. See the ["Instruction Set Description"](#) for detailed information.

### • Bit 2 – N: Negative Flag

The Negative Flag N indicates a negative result in an arithmetic or logic operation. See the ["Instruction Set Description"](#) for detailed information.

## • Bit 1 – Z: Zero Flag

The Zero Flag Z indicates a zero result in an arithmetic or logic operation. See the “[Instruction Set Description](#)” for detailed information.

## • Bit 0 – C: Carry Flag

The Carry Flag C indicates a Carry in an arithmetic or logic operation. See the “[Instruction Set Description](#)” for detailed information.

## General Purpose Register File

The Register File is optimized for the AVR Enhanced RISC instruction set. In order to achieve the required performance and flexibility, the following input/output schemes are supported by the Register File:

- One 8-bit output operand and one 8-bit result input
- Two 8-bit output operands and one 8-bit result input
- Two 8-bit output operands and one 16-bit result input
- One 16-bit output operand and one 16-bit result input

[Figure 3](#) shows the structure of the 32 general purpose working registers in the CPU.

**Figure 3.** AVR CPU General Purpose Working Registers

	7	0	Addr.	
General Purpose Working Registers	R0		0x00	
	R1		0x01	
	R2		0x02	
	...			
	R13		0x0D	
	R14		0x0E	
	R15		0x0F	
	R16		0x10	
	R17		0x11	
	...			
	R26		0x1A	X-register Low Byte
	R27		0x1B	X-register High Byte
	R28		0x1C	Y-register Low Byte
	R29		0x1D	Y-register High Byte
	R30		0x1E	Z-register Low Byte
	R31		0x1F	Z-register High Byte

Most of the instructions operating on the Register File have direct access to all registers, and most of them are single cycle instructions.

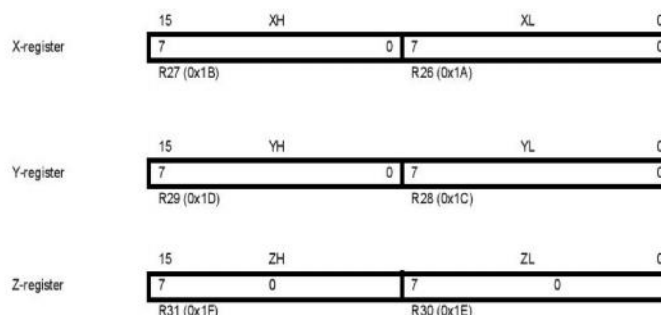
As shown in [Figure 3](#), each register is also assigned a Data memory address, mapping them directly into the first 32 locations of the user Data Space. Although not being physically implemented as SRAM locations, this memory organization provides great flexibility in access of the registers, as the X-pointer, Y-pointer, and Z-pointer Registers can be set to index any register in the file.



## The X-register, Y-register and Z-register

The registers R26..R31 have some added functions to their general purpose usage. These registers are 16-bit address pointers for indirect addressing of the Data Space. The three indirect address registers X, Y and Z are defined as described in [Figure 4](#).

**Figure 4.** The X-register, Y-register and Z-Register



In the different addressing modes these address registers have functions as fixed displacement, automatic increment, and automatic decrement (see the [Instruction Set Reference](#) for details).

## Stack Pointer

The Stack is mainly used for storing temporary data, for storing local variables and for storing return addresses after interrupts and subroutine calls. The Stack Pointer Register always points to the top of the Stack. Note that the Stack is implemented as growing from higher memory locations to lower memory locations. This implies that a Stack PUSH command decreases the Stack Pointer.

The Stack Pointer points to the data SRAM Stack area where the Subroutine and Interrupt Stacks are located. This Stack space in the data SRAM must be defined by the program before any subroutine calls are executed or interrupts are enabled. The Stack Pointer must be set to point above 0x60. The Stack Pointer is decremented by one when data is pushed onto the Stack with the PUSH instruction, and it is decremented by two when the return address is pushed onto the Stack with subroutine call or interrupt. The Stack Pointer is incremented by one when data is popped from the Stack with the POP instruction, and it is incremented by two when address is popped from the Stack with return from subroutine RET or return from interrupt RETI.

The AVR Stack Pointer is implemented as two 8-bit registers in the I/O space. The number of bits actually used is implementation dependent. Note that the data space in some implementations of the AVR architecture is so small that only SPL is needed. In this case, the SPH Register will not be present.

Bit	15	14	13	12	11	10	9	8	
	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	SPH
	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	SPL
	7	6	5	4	3	2	1	0	
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

## Instruction Execution Timing

This section describes the general access timing concepts for instruction execution. The Atmel®AVR® CPU is driven by the CPU clock  $clk_{CPU}$ , directly generated from the selected clock source for the chip. No internal clock division is used.

Figure 5 shows the parallel instruction fetches and instruction executions enabled by the Harvard architecture and the fast-access Register File concept. This is the basic pipelining concept to obtain up to 1MIPS per MHz with the corresponding unique results for functions per cost, functions per clocks, and functions per power-unit.

**Figure 5. The Parallel Instruction Fetches and Instruction Executions**

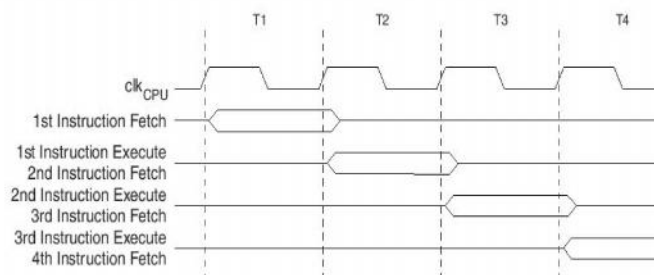
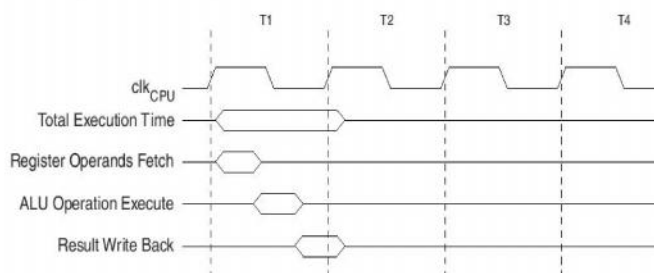


Figure 6 shows the internal timing concept for the Register File. In a single clock cycle an ALU operation using two register operands is executed, and the result is stored back to the destination register.

**Figure 6. Single Cycle ALU Operation**



## Reset and Interrupt Handling

The Atmel®AVR® provides several different interrupt sources. These interrupts and the separate Reset Vector each have a separate Program Vector in the Program memory space. All interrupts are assigned individual enable bits which must be written logic one together with the Global Interrupt Enable bit in the Status Register in order to enable the interrupt. Depending on the Program Counter value, interrupts may be automatically disabled when Boot Lock Bits BLB02 or BLB12 are programmed. This feature improves software security. See the section ["Memory Programming" on page 215](#) for details.

The lowest addresses in the Program memory space are by default defined as the Reset and Interrupt Vectors. The complete list of Vectors is shown in ["Interrupts" on page 46](#). The list also determines the priority levels of the different interrupts. The lower the address the higher is the priority level. RESET has the highest priority, and next is INT0 – the External Interrupt Request 0. The Interrupt Vectors can be moved to the start of the boot Flash section by setting the Interrupt Vector Select (IVSEL) bit in the General Interrupt Control Register (GICR). Refer to ["Interrupts" on page 46](#) for more information. The Reset Vector can also be moved to the start of the boot Flash section by programming the BOOTRST Fuse, see ["Boot Loader Support – Read-While-Write Self-Programming" on page 202](#).



When an interrupt occurs, the Global Interrupt Enable I-bit is cleared and all interrupts are disabled. The user software can write logic one to the I-bit to enable nested interrupts. All enabled interrupts can then interrupt the current interrupt routine. The I-bit is automatically set when a Return from Interrupt instruction – RETI – is executed.

There are basically two types of interrupts. The first type is triggered by an event that sets the Interrupt Flag. For these interrupts, the Program Counter is vectored to the actual Interrupt Vector in order to execute the interrupt handling routine, and hardware clears the corresponding Interrupt Flag. Interrupt Flags can also be cleared by writing a logic one to the flag bit position(s) to be cleared. If an interrupt condition occurs while the corresponding interrupt enable bit is cleared, the Interrupt Flag will be set and remembered until the interrupt is enabled, or the flag is cleared by software. Similarly, if one or more interrupt conditions occur while the global interrupt enable bit is cleared, the corresponding Interrupt Flag(s) will be set and remembered until the global interrupt enable bit is set, and will then be executed by order of priority.

The second type of interrupts will trigger as long as the interrupt condition is present. These interrupts do not necessarily have Interrupt Flags. If the interrupt condition disappears before the interrupt is enabled, the interrupt will not be triggered.

When the AVR exits from an interrupt, it will always return to the main program and execute one more instruction before any pending interrupt is served.

Note that the Status Register is not automatically stored when entering an interrupt routine, nor restored when returning from an interrupt routine. This must be handled by software.

When using the CLI instruction to disable interrupts, the interrupts will be immediately disabled. No interrupt will be executed after the CLI instruction, even if it occurs simultaneously with the CLI instruction. The following example shows how this can be used to avoid interrupts during the timed EEPROM write sequence.

## Assembly Code Example

```
in r16, SREG      ; store SREG value
cli              ; disable interrupts during timed sequence
sbi EECR, EEMWE   ; start EEPROM write
sbi EECR, EWE     ; start EEPROM write
out SREG, r16     ; restore SREG value (I-bit)
```

## C Code Example

```
char cSREG;
cSREG = SREG; /* store SREG value */
/* disable interrupts during timed sequence */
_cli();
EECR |= (1<<EEMWE); /* start EEPROM write */
EECR |= (1<<EWE);
SREG = cSREG; /* restore SREG value (I-bit) */
```

When using the SEI instruction to enable interrupts, the instruction following SEI will be executed before any pending interrupts, as shown in the following example.

Assembly Code Example
<pre>sei ; set global interrupt enable sleep; enter sleep, waiting for interrupt ; note: will enter sleep before any pending ; interrupt(s)</pre>
C Code Example
<pre>_SEI(); /* set global interrupt enable */ _SLEEP(); /* enter sleep, waiting for interrupt */ /* note: will enter sleep before any pending interrupt(s) */</pre>

## Interrupt Response Time

The interrupt execution response for all the enabled Atmel®AVR® interrupts is four clock cycles minimum. After four clock cycles, the Program Vector address for the actual interrupt handling routine is executed. During this 4-clock cycle period, the Program Counter is pushed onto the Stack. The Vector is normally a jump to the interrupt routine, and this jump takes three clock cycles. If an interrupt occurs during execution of a multi-cycle instruction, this instruction is completed before the interrupt is served. If an interrupt occurs when the MCU is in sleep mode, the interrupt execution response time is increased by four clock cycles. This increase comes in addition to the start-up time from the selected sleep mode.

A return from an interrupt handling routine takes four clock cycles. During these four clock cycles, the Program Counter (2 bytes) is popped back from the Stack, the Stack Pointer is incremented by 2, and the I-bit in SREG is set.

**AOSONG**

Temperature and humidity module  
DHT11 Product Manual



[www.aosong.com](http://www.aosong.com)

## 1、Product Overview

DHT11 digital temperature and humidity sensor is a composite Sensor contains a calibrated digital signal output of the temperature and humidity. Application of a dedicated digital modules collection technology and the temperature and humidity sensing technology, to ensure that the product has high reliability and excellent long-term stability. The sensor includes a resistive sense of wet components and an NTC temperature measurement devices, and connected with a high-performance 8-bit microcontroller.



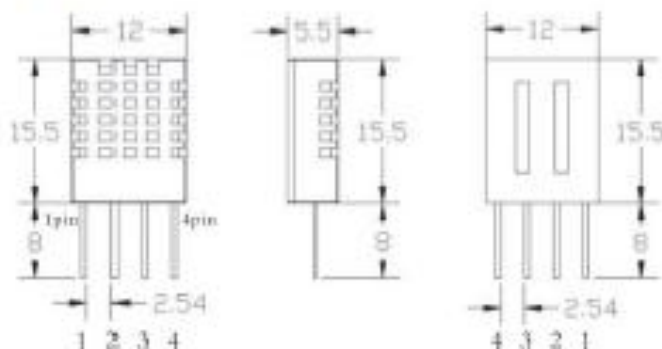
## 2、Applications

HVAC, dehumidifier, testing and inspection equipment, consumer goods, automotive, automatic control, data loggers, weather stations, home appliances, humidity regulator, medical and other humidity measurement and control.

## 3、Features

Low cost, long-term stability, relative humidity and temperature measurement, excellent quality, fast response, strong anti-interference ability, long distance signal transmission, digital signal output, and precise calibration.

## 4、Dimensions (unit: mm)



## 5. Product parameters

### Relative humidity

Resolution: 16Bit

Repeatability:  $\pm 1\%$  RH

Accuracy: At 25°C  $\pm 5\%$  RH

Interchangeability: fully interchangeable

Response time: 1 / e (63%) of 25°C 6s

1m / s air 6s

Hysteresis:  $< \pm 0.3\%$  RH

Long-term stability:  $< \pm 0.5\%$  RH / yr in

### Temperature

Resolution: 16Bit

Repeatability:  $\pm 0.2^\circ\text{C}$

Range: At 25°C  $\pm 2^\circ\text{C}$

Response time: 1 / e (63%) 10S

### Electrical Characteristics

Power supply: DC 3.5 ~ 5.5V

Supply Current: measurement 0.3mA standby 60 $\mu$  A

Sampling period: more than 2 seconds

### Pin Description

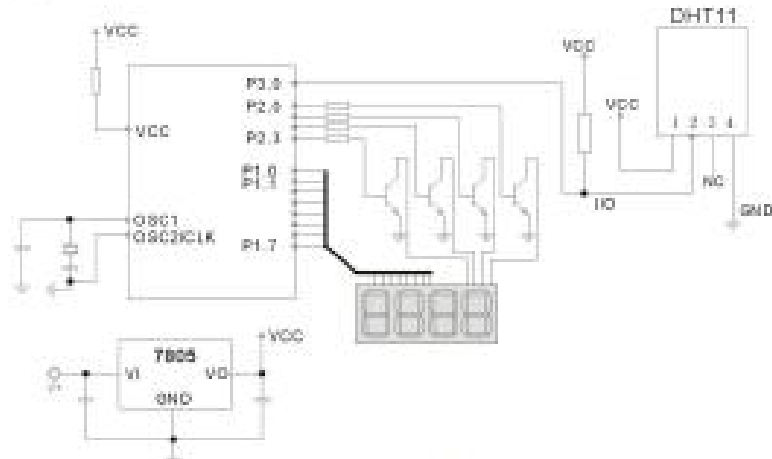
1, the VDD power supply 3.5 ~ 5.5V DC

2 DATA serial data, a single bus

3, NC, empty pin

4, GND ground, the negative power

## 6. Typical circuit



Microprocessor and DHT11 of connection typical application circuit as shown above, DATA pull the microprocessor I/O ports are connected.

1. Typical application circuit recommended in the short cable length of 20 meters on the 5.1K pull-up resistor, the resistance of greater than 20 meters under the pull-up resistor on the lower of the actual situation.

2. When using a 3.5V voltage supply cable length shall not be greater than 20cm. Otherwise, the line voltage drop will cause the sensor power supply shortage, caused by measurement error.

3. Each read out the temperature and humidity values are the results of the last measurement For real-time data, sequential read twice, but is not recommended to repeatedly read the sensors, each read sensor interval is greater than 5 seconds can be obtained accurate data.

## 7. Serial communication instructions (single-wire bi-directional)

### Single bus Description

DHT11 uses a simplified single-bus communication. Single bus that only one data line, the system of data exchange, control by a single bus to complete. Device (master or slave) through an open-drain or tri-state port connected to the data line to allow the device does not send data to release the bus, while other devices use the bus; single bus usually require an external one about 5.1kΩ pull-up resistor, so that when the bus is idle, its status is high. Because they are the master-slave structure, and only when the host calls the slave, the slave can answer, the host access devices must strictly follow the single-bus sequence, if the chaotic sequence, the device will not respond to the host.

### Single bus to transfer data defined

DATA For communication and synchronization between the microprocessor and DHT11, single-bus data format, a transmission of 40 data, the high first-out.

### Data format:

The 8bit humidity integer data + 8bit the Humidity decimal data +8 bit temperature integer data + 8bit fractional temperature data +8 bit parity bit.

### ①Parity bit data definition

"8bit humidity integer data + 8bit humidity decimal data +8 bit temperature integer data + 8bit temperature fractional data" 8bit checksum is equal to the results of the last eight.

Example 1: 40 data is received:

<u>0011 0101</u>	<u>0000 0000</u>	<u>0001 1000</u>	<u>0000 0000</u>	<u>0100 1101</u>
High humidity 8	Low humidity 8	High temp. 8	Low temp. 8	Parity bit

Calculate:

$0011\ 0101 + 0000\ 0000 + 0001\ 1000 + 0000\ 0000 = 0100\ 1101$

Received data is correct;

Humidity:  $0011\ 0101 = 35H = 53\%RH$

Temperature:  $0001\ 1000 = 18H = 24^{\circ}C$

Example 2: 40 data is received:

<u>0011 0101</u>	<u>0000 0000</u>	<u>0001 1000</u>	<u>0000 0000</u>	<u>0100 1001</u>
High humidity 8	Low humidity 8	High temp. 8	Low temp. 8	Parity bit

Calculate:

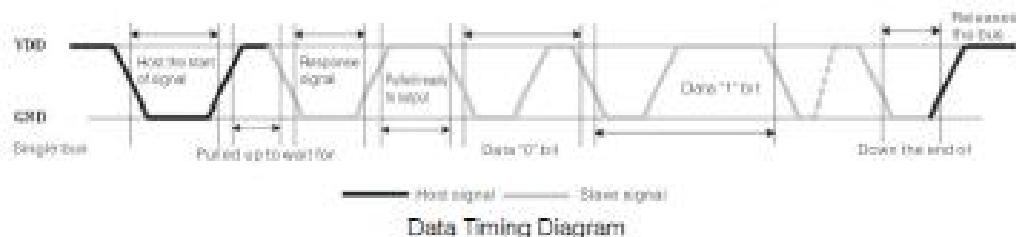
$0011\ 0101 + 0000\ 0000 + 0001\ 1000 + 0000\ 0000 = 0100\ 1101$

$01001101 \neq 0100\ 1001$

The received data is not correct, give up, to re-receive data.

### ②Data Timing Diagram

User host (MCU) to send a signal, DHT11 converted from low-power mode to high-speed mode, until the host began to signal the end of the DHT11 send a response signal to send 40bit data, and trigger a letter collection. The signal is sent as shown.



**Note:** The host reads the temperature and humidity data from DHT11 always the last measured value, such as twice the measured interval of time is very long, continuous read twice to the second value of real-time temperature and humidity values.

### ○Peripherals read steps

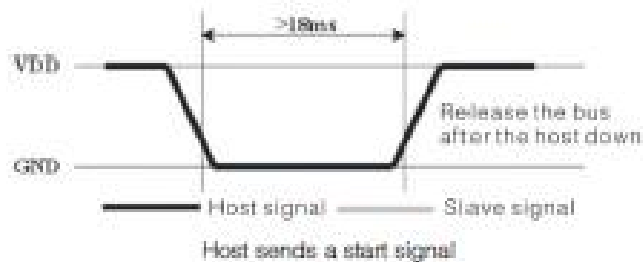
Communication between the master and slave can be done through the following steps (peripherals (such as microprocessors) read DHT11 the data of steps).

#### Step 1:

After power on DHT11 (DHT11 on after power to wait 1S across the unstable state during this period can not send any instruction), the test environment temperature and humidity data, and record the data, while DHT11 the DATA data lines pulled by pull-up resistor has been to maintain high; the DHT11 the DATA pin is in input state, the moment of detection of external signals.

#### Step 2:

Microprocessor I / O set to output at the same time output low, and low hold time can not be less than 18ms, then the microprocessor I / O is set to input state, due to the pull-up resistor, a microprocessor / O DHT11 the DATA data lines also will be high, waiting DHT11 to answer signal, send the signal as shown:



#### Step 3:

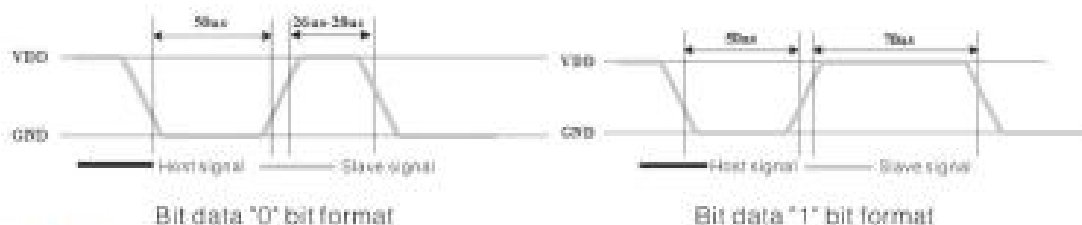
DATA pin is detected to an external signal of DHT11 low, waiting for external signal low and the delay DHT11 DATA pin in the output state, the output low of 80 microseconds as the response signal, followed by the output of 80 micro-seconds of high notification peripheral is ready to receive data, the microprocessor I / O at this time in the input state is detected the I / O low (DHT11 response signal), wait 80 microseconds high data receiving and sending signals as shown:



#### Step 4:

Output by DHT11 the DATA pin 40, the microprocessor receives 40 data bits of data '0' format: the low level of 50 microseconds and 26-28 microseconds according to the changes in the I / O level level, bit data '1' format: the high level of low plus, 50 microseconds to 70 microseconds. Bit data '0', '1' signal format as shown:





## End signal:

Continue to output the low 50 microseconds after DHT11 the DATA pin output 40 data, and changed the input state, along with pull-up resistor goes high. But DHT11 internal re-test environmental temperature and humidity data, and record the data, waiting for the arrival of the external signal.

## 8. Application of information

### 1. Work and storage conditions

Outside the sensor the proposed scope of work may lead to temporary drift of the signal up to 300%RH. Return to normal working conditions, sensor calibration status will slowly toward recovery. To speed up the recovery process may refer to "resume processing". Prolonged use of non-normal operating conditions, will accelerate the aging of the product.

Avoid placing the components on the long-term condensation and dry environment, as well as the following environment.

A, salt spray

B, acidic or oxidizing gases such as sulfur dioxide, hydrochloric acid

Recommended storage environment

Temperature: 10 ~ 40 °C Humidity: 60% RH or less

### 2. The impact of exposure to chemicals

The capacitive humidity sensor has a layer by chemical vapor interference, the proliferation of chemicals in the sensing layer may lead to drift and decreased sensitivity of the measured values. In a pure environment, contaminants will slowly be released. Resume processing as described below will accelerate this process. The high concentration of chemical pollution (such as ethanol) will lead to the complete damage of the sensitive layer of the sensor.

### 3. The temperature influence

Relative humidity of the gas to a large extent dependent on temperature. Therefore, in the measurement of humidity, should be to ensure that the work of the humidity sensor at the same temperature. With the release of heat of electronic components share a printed circuit board, the installation should be as far as possible the sensor away from the electronic components and mounted below the heat source, while maintaining good ventilation of the enclosure. To reduce the thermal conductivity sensor and printed circuit board copper plating should be the smallest possible, and leaving a gap between the two.

### 4. Light impact

Prolonged exposure to sunlight or strong ultraviolet radiation, and degrade performance.

### 5. Resume processing

Placed under extreme working conditions or chemical vapor sensor, which allows it to return to the status of calibration by the following handler. Maintain two hours in the humidity conditions of 45°C and <10% R.H (dry); followed by 20~30°C and > 70% R.H humidity conditions to maintain more than five hours.

### 6. Wiring precautions

The quality of the signal wire will affect the quality of the voltage output, it is recommended to use high quality shielded cable.

### 7. Welding information

Manual welding, in the maximum temperature of 300°C under the conditions of contact time shall be less than 3 seconds.

### 8. Product upgrades

Details, please the consultation Aosong electronics department.

### 9. The license agreement

Without the prior written permission of the copyright holder, shall not in any form or by any means, electronic or mechanical (including photocopying), copy any part of this manual, nor shall its contents be communicated to a third party. The contents are subject to change without notice.

The Company and third parties have ownership of the software, the user may use only signed a contract or software license.

### 10. Warnings and personal injury

This product is not applied to the safety or emergency stop devices, as well as the failure of the product may result in injury to any other application, unless a particular purpose or use authorized. Installation, handling, use or maintenance of the product refer to product data sheets and application notes. Failure to comply with this recommendation may result in death and serious personal injury. The Company will bear all damages resulting personal injury or death, and waive any claims that the resulting subsidiary company managers and employees and agents, distributors, etc. that may arise, including: a variety of costs, compensation costs, attorneys' fees, and so on.

### 11. Quality Assurance

The company and its direct purchaser of the product quality guarantee period of three months (from the date of delivery). Publishes the technical specifications of the product data sheet shall prevail. Within the warranty period, the product was confirmed that the quality is really defective, the company will provide free repair or replacement. The user must satisfy the following conditions:

- ① The product is found defective within 14 days written notice to the Company;
- ② The product shall be paid by mail back to the company;
- ③ The product should be within the warranty period.

The Company is only responsible for those used in the occasion of the technical condition of the product defective product. Without any guarantee, warranty or written statement of its products used in special applications. Company for its products applied to the reliability of the product or circuit does not make any commitment.

Aosong(Guangzhou) Electronics Co.,Ltd. TEL: 020-36042809 / 36380552 [www.aosong.com](http://www.aosong.com)

- 7 -



## **KYL-200U Micro Power Wireless Transceiver Data Module**



### **Shenzhen KYL Communication Equipment Co., Ltd**

**Address:** Unit 5-7, Building3, Huangdu Square, South of Exhibition Center, Yitian Road, Futian District, Shenzhen Guangdong China

**Tel:** +0086-755-83410815 / 82943662

**Fax:** +0086-755-83408785

**Skype:** lxmxlosia

**Yahoo messenger:** losia2012@yahoo.com

**MSN:** kylcom01.com@hotmail.com

**E\_mail:** [sales01@rf-data.com](mailto:sales01@rf-data.com) ; [sales02@rf-data.com](mailto:sales02@rf-data.com)

**Website:** <http://www.rf-data.com>

**Fax:** +0086-755-83408785    [kyl03losia@gmail.com](mailto:kyl03losia@gmail.com)    [www.rf-data.com](http://www.rf-data.com)

1

## **Catalogue**

<b>I. About KYL-200U .....</b>	<b>3</b>
<b>II. Features of KYL-200U .....</b>	<b>3</b>
<b>III. Applications of KYL-200U.....</b>	<b>4</b>
<b>IV. How to use KYL-200U.....</b>	<b>4</b>
<b>V. Networking Application of KYL-200U.....</b>	<b>14</b>
<b>VI. Technical specification of KYL-200U.....</b>	<b>14</b>
<b>VII. Description of type.....</b>	<b>15</b>

Before using the product please read the use manual carefully.

Any question in technical, you can contact us.

Tel: +0086-755-83410815.

## **I: About KYL-200U**

KYL-200U, the micro power wireless transceiver data module is used as the wireless data transceiver in short-ranges, with the small size, weight and power consumption and good stability and reliability. Narrowband low power UHF wireless data transmitters and receivers with channel spacing as low as 25 KHz.

## **II: Features:**

### **I. Features of KYL-200U**

1. Low power transmission with 10mW, 50~100mW is available if needed.
2. With 433MHz as factory default carrier frequency, but 400-470MHz, 868MHz, 915MHz is available if needed.
3. High anti- interference and Low BER (Bit error Rate)  
Based on the Gaussian Frequency Shift Keying (GFSK) modulation, the high-efficiency forward error correction channel encoding technology is used to enhance data's resistance to both transient interference and random interference and the actual bit error rate of  $10^{-5} \sim 10^{-6}$  can be achieved when channel bit error rate is  $10^{-3}$ .
4. Long Transmission Distance  
Within the visible range, when the height of antenna is higher than 2m and The Bit Error Rate (BER) is  $10^{-5}$ , the reliable transmission distances respectively is 1000m @1200bps, and is 600m@9600bps.
5. Transparent data transmission  
Transparent data interface used in transceivers is for meeting many standard or nonstandard user protocols. Any false data generated in air can be filtrated automatically (What has been received is exactly what has been transmitted).
6. Multi-channels  
KYL-200U transceivers offer 8 channels, if needed, 16 or 32 channels are available to satisfy various configuration of communication under user's demand at the same time.
7. 2 ports with three connection methods  
KYL-200U transceivers provide 2 ports with three connections, ports: a UART interface of TTL level, a non-standard RS-232 port and a non-standard RS-485 port.
8. Big data buffer area  
With optional interface baud rate: 1200/4800/9600/19200bps and 8N1/8E1/8O1 data format (set by user), the transceiver can transmit unlimited data frames with flexible user program.  
Note: the RF data rate is only settled down : delivery, please inform our sales when placing your orders.

**Fax: +0086-755-83408785    kyl03loala@gmail.com    www.rf-data.com**

3

9. Intelligent data control and no any extra programs required  
Even for half duplex communication, no any excessive programs required. All RF system data transmission/reception and other On-the-Fly conversion and control are performed by KYL-200U transceivers automatically.
10. Lower power consumption & Sleep function  
With +5V power, the receiving current is less than 20mA, the transmitting current is less than 40mA, and the sleeping current is less than 20uA.
11. High reliability, small and light  
By using monolithic radio-frequency integrated circuit and single-chip MCU, the transceivers have less peripheral circuit s, high reliability, and low failure rate.
12. More options of configurable antennas for user different applications.
13. Watchdog monitor  
Watchdog monitors the inner function, so it can change the traditional product structure and also improve the reliability of our modules.

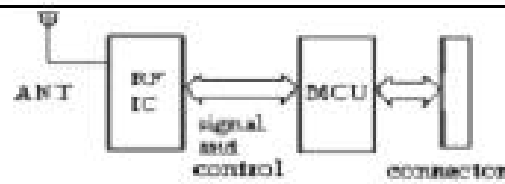
### **III: Application of KYL-200U:**

KYL-200U the micro power wireless transceiver data module is suitable for:

- \* AMR Automatic Meter Reading
- \* Wireless alarm and security systems
- \* Building automation, security, wireless monitoring and control of room equipment, Access Control System;
- \* Wireless data transmission, automatic data collection system;
- \* Radio modem can be used for Sports training & competition;
- \* Wireless dishes ordering;
- \* Wireless POS, PDA wireless smart terminal;
- \* RF modem can be used for Electronic bus station and intelligent traffic;
- \* RF transmitter Wireless electronic display screen and queuing machine;
- \* Wireless telemetry Charging for parking, parking lot;
- \* Wireless modem Automobile inspection and four-wheel orientation;
- \* Wireless sensor Industrial wireless remote control and air conditioning remote controller;
- \* Data communication used for railway, oil field, dock and army.
- \* LED display in thruway and public places
- \* Point to multi-point wireless network, wireless on-the-spot bus and automatic data collection system;

### **IV: How to use the KYL-200U**

KYL-200U provide RS-232, RS-485 and UART/TTL level interface port for direct connection with PC, RS485 devices, monolithic processors and other UART components kinds of applications. The schematic diagram is shown below:



KYL-200U Principle map

### 1. Power supply

The factory default setting is +5V(3.3-3.6V will inform when placing the order). By using better ripple factor, KYL-200U transceivers can also share power supply with other equipment. If possible, a voltage-stabilizing chip with 5V voltage is more recommended as the only power supply than Switch power supply. But if only switch power supply available, the jam by switch pulse to the transceivers should be avoided. In addition, the reliable grounding must be used if there is other device in the system equipment. In case of failing to connect with the ground, it can form its own grounding but must be absolutely separated from the municipal electric supply.

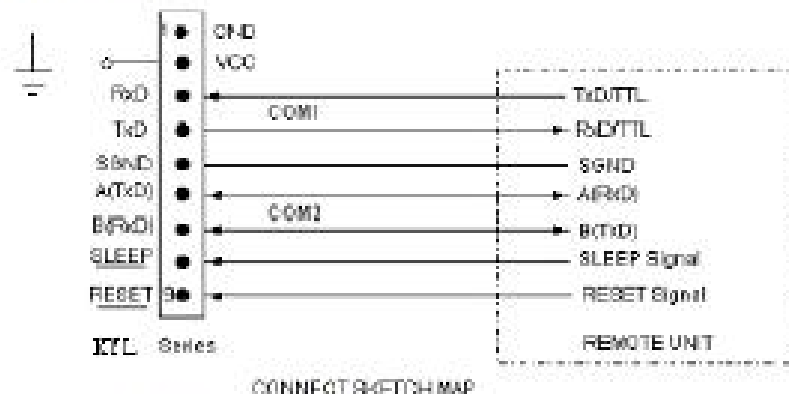
### 2. Connection Definition with terminal

KYL-200U transceivers supply one 9-pin connector (JP1), their definitions and connection methods with terminals are shown in Table 1.

**Table 1: JP1 Pin Definitions and connection methods**

Pin No.	Signal Name	Function	Level	Connection with terminal	Remarks
1	GND	Grounding of power supply		Ground	
2	Vcc	Power supply DC	5V		
3	RxD/TTL	Serial data input to the transceiver	TTL	TxD	
4	TxD/TTL	Transmitted data out of the transceiver	TTL	RxD	
5	SGND	Signal			
6	A (TxD)	A of RS-485(TxD of RS-232)		A(RxD)	
7	B (RxD)	B of RS-485(RxD of RS-232)		B(TxD)	
8	SLEEP	Sleep control (input)	TTL	Sleep signal	High level sleep
9	RESET	Reset signal(input)	TTL		Negative pulse reset

### 3. The connection schematic diagram between computer and our RF module



### 4. Setting of channel, interface, and data format

Before using KYL-200U, the user needs to make simple configuration based on its own needs to determine the channel, interface mode and data format. The user can change or view the module's interface baud rate, channel and address code. Parameter setting or reading as per the testing software KYLCOM.exe in the PC (in products box). And the configuration is as follows:

#### i. Channel configuration:

Channel No.	Frequency	Channel No.	Frequency
1	429.0325MHZ	5	433.0325MHZ
2	430.0325MHZ	6	434.0325MHZ
3	431.0325MHZ	7	435.0325MHZ
4	432.0325MHZ	8	436.0325MHZ

**Note: the frequency points corresponding to each channel can be adjusted based on the user's needs.**

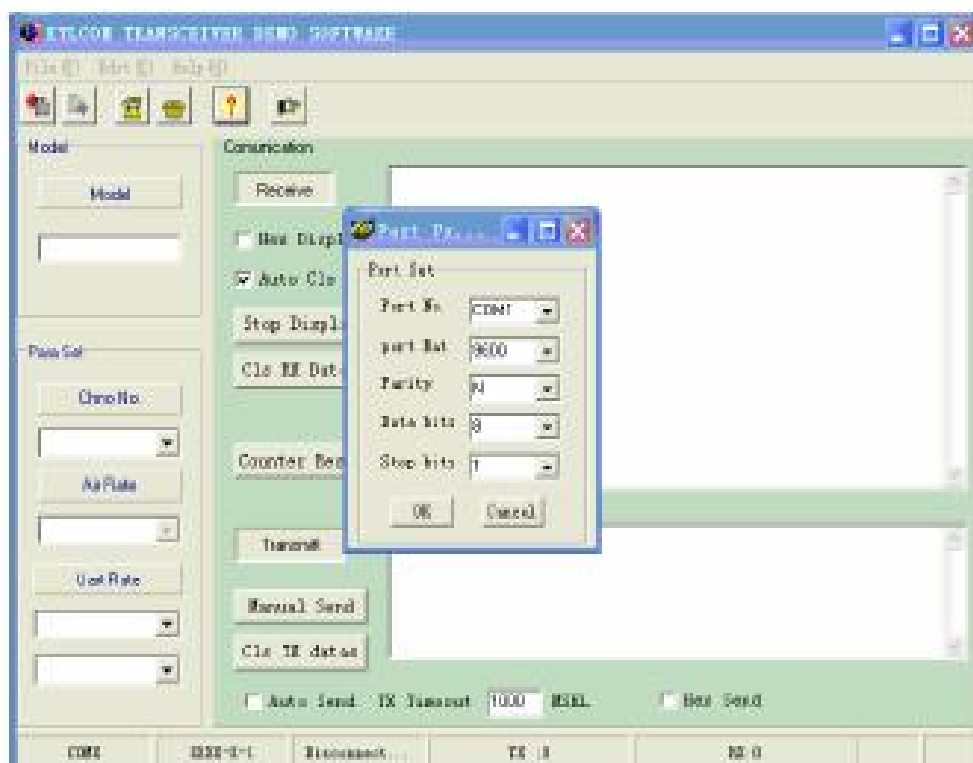
#### ii. The schematic diagram of setting the parameter as follows:

- Connect the PC and module with RS232 data cable.
- Open the "KYLCOM" PC software, select "English" as follows:

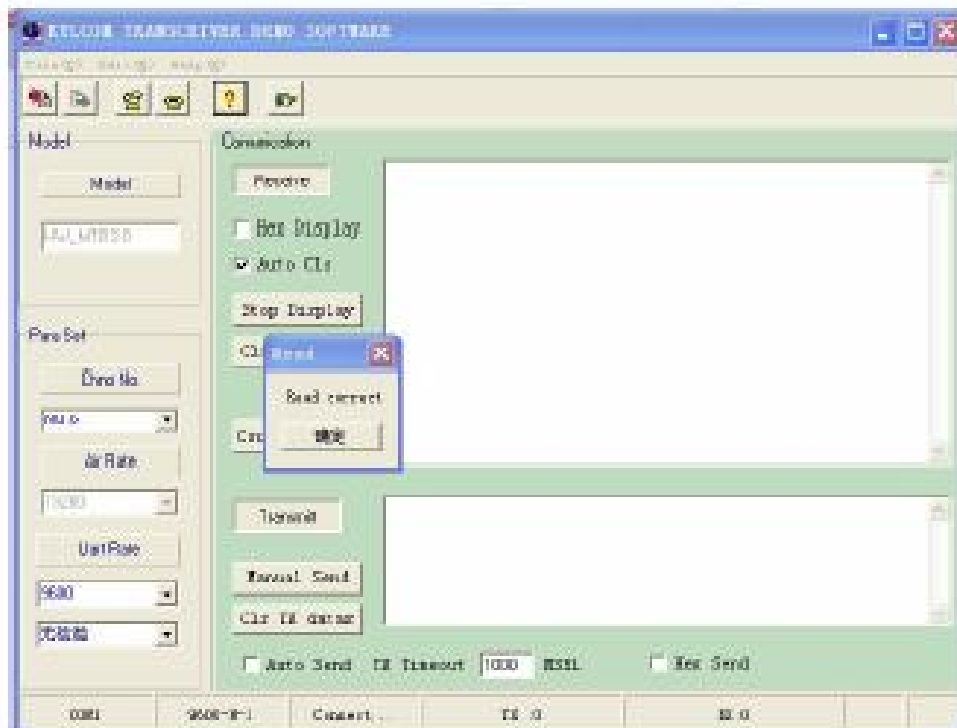




- c. Click "open port", then will open a small window as the following picture, clients will write the parameters in this windows.

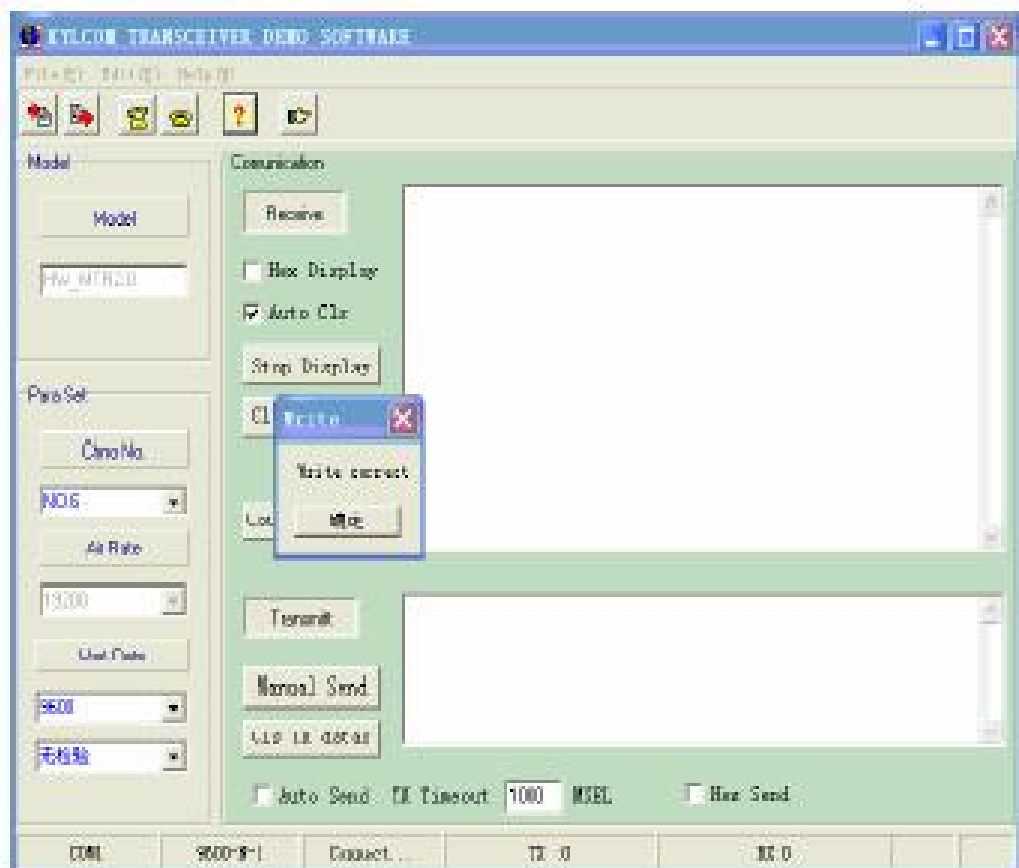


- d. Click "read paras".  
"Read correct" show your connection between KYL module and PC is correct.



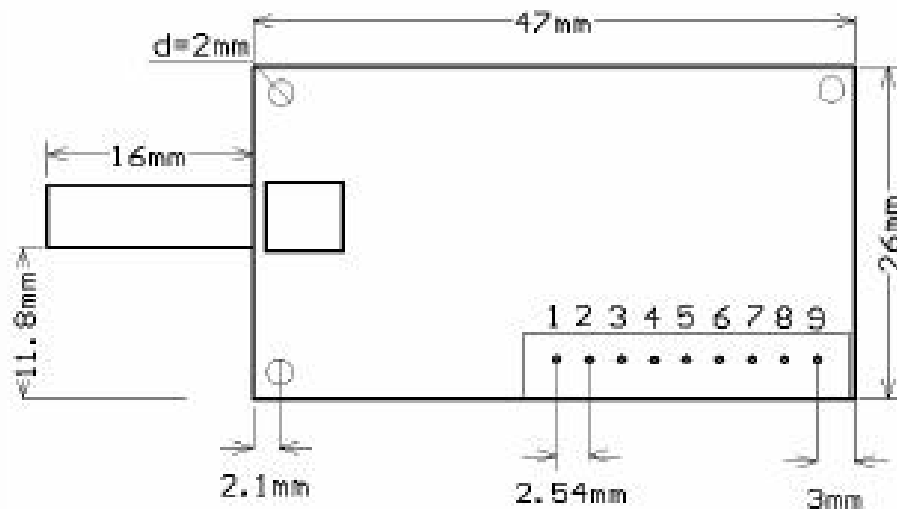
If you want to change some technical parameter, you need do as follows.

- e. User can change the parameter such as channel and interface rate and Parity mode according to actual demand through the button of the left window. After writing the parameters in the left catalogue, click the "write para", then will show "write correct". So you have finished the process for changing technical parameter.



**Note:** The RF data rate(Air Rate) can not be set by user, it is confirmed when placing orders and set before delivery. So please confirm the RF data rate when placing the order.

##### 5. Installation dimension:



#### 6. The Function-indicator light

- The LED indicator light will glitter red for 0.5S once after switched on.
- The LED indicator light will glitter green continually while receiving data from air.
- The LED indicator light will glitter red continuously when transmitting data into air after receiving from COM.
- If the module enables the sleep function, LED indicator light is always dark.

#### 7. Supported protocol and Transmit capability

KYL-200U standard transceivers offer transparent protocol to support various applications and protocols of users. If the user needs to decrease his cost or ease the workload of terminal CPU, we can add other specific functions based on the transparent protocol, such as addressing, data acquisition, command interpretation, etc.

#### 8. Sleep function instruction:

Due to reducing more consumption, KYL-200U transceivers support Sleep function. In sleep mode, the current consumption is less than 20uA.

##### a. How to use the Sleep function:

The Pin8 'SLP' in JP1 is the signal of sleep control. In high power level, transceiver stays in working mode. In low power level ( $<0.5V$ ), transceiver stays in sleep mode. The SLP signal can convert transceiver from working to sleep mode in 1ms after falling edge. If the Sleep signal arrives when the transceiver is transmitting data, the module will enter sleep mode after finishing transmission. From sleep mode to working, it takes the transceiver 1ms after rising edge. To disable the opened sleep function of KYL-200U, the SLP (SLEEP) pin should be definitely connected with 0 or ground.

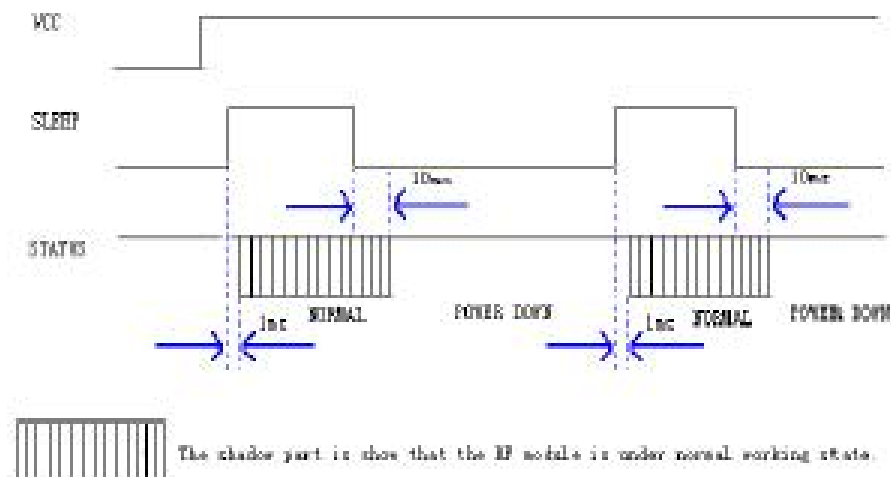
##### b. Attentions about use of sleep function:

When the sleep function enabled, any supply glitches, such as switch dithering,

fire striking or quick switching on and off, could cause the transceiver to be switched to the wrong sleep mode.

After switching on, users can avoid this error by making a compulsive restoration once after the CPU delays 100ms.

**Sleep Timing Diagram:**

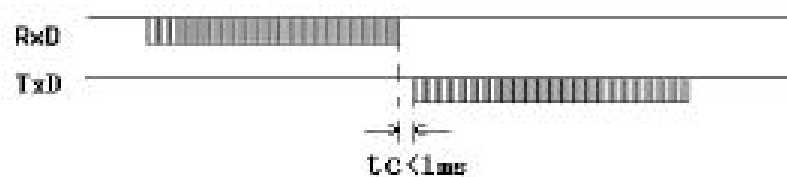


**9. The attentions of data transmission**

- a. The delay time ( $t_c$ ) of conversion between transmitting and receiving is less than 1ms.

**Timing diagram:**

KYL SERIES

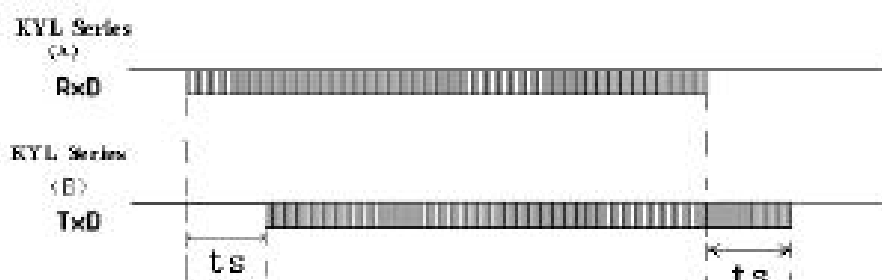


- b. The delay time of transceivers between the first bit sent by TxD to the first bit received by RxD.

Due to a data processing will be made on user's data by KYL-200U transceiver using FEC (Forward Error Correction) or other correction algorithm, when RxD of a KYL-200U transceiver 'A' receives the data, then transmits it, the other one transceiver 'B' will have a delay ( $t_s$ ) to receive and transmit by TxD. Different RF data rate causes different delay time. Please see the specific delay time below:

RF Data Rate (bps)	Delay Ts(mS)	RF Data Rate (bps)	Delay Ts(mS)
1200	90	9600	16
2400	48	19200	10
4800	30		

### Timing diagram:



#### c. Error dealing procedure:

To enhance the reliability and stability of user's systems, a verify bit or a Cyclic Redundancy Check (CRC) mode is highly recommended to resent the wrong information while using KYL-200U modules.

#### d. Large-number data transmission

In theory, when the interface data rate is faster than the RF data rate, KYL-200U transceivers can sent unlimited-long data package, but any long packages more than 120B are not recommended. The length of each package should be between 60~100B. We also recommend user to resent the wrong information using Automatic Error Request Equipment (ARQ).

#### The analyzing as below:

What if the actual transmission BER (Bit Error Rate) is  $10^{-4}$ , 1 packet with 1KB data, which is about 10-thousand bits, is sent, theoretically, at least 1 bit will be received wrongly, then the 1KB information will never be received correctly. But if we package the data into 10 packets with 100B for each, when all 10 packets are sent, there will be only 1 packet wrong according to this probability. After that, resend this wrong packet using ARQ mode. So by resending one more packet and the efficiency rate is reduced 10%, all data will be absolutely received correctly.

### 10. Antenna configuration:

Many appropriate antennas for low power RF modules are selected for meeting different user antenna configurations. Please ask our Sales office for further

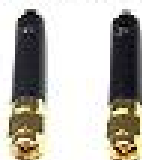
information about the antenna's dimension and performance.

a. Helical SMA antennas

KYL-ANT-433-10-SMA: 100mm helical SMA antennas with high gain and low cost, reach a long distance.



KYL-ANT-433-3-SMA: 28mm SMA helical antenna with magnetic core, small size, beautiful appearance (like the mobile phone antenna)

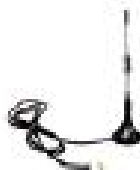


KYL-ANT-433-10-ZSMA: 100mm SMA helical antenna with folding joint, easy for adjusting the direction and fixing.



b. Magnetic vehicle antenna

KYL-ANT-O433S-300H1.5-SMA: include magnetic antenna base, suitable for equipment with metal shell, easy for fixing, effective to enhance the transmission distance.



More kinds of antenna, please browse our website:

[http://www.rf-data.com/Products\\_list\\_en.asp?classid=163](http://www.rf-data.com/Products_list_en.asp?classid=163)

If the user has special demands on antennas, we can design and produce for them specially.

## V. Networking Application of KYL-200U

The communication channel of KYL-200U is half duplex, which is most suitable for the communication mode of point to multi-point. Under this mode, one master station must be set, and all of the rest are slave stations. A unique address

**Fax: +0086-755-83408785** [kyl03loela@gmail.com](mailto:kyl03loela@gmail.com) [www.rf-data.com](http://www.rf-data.com)

13

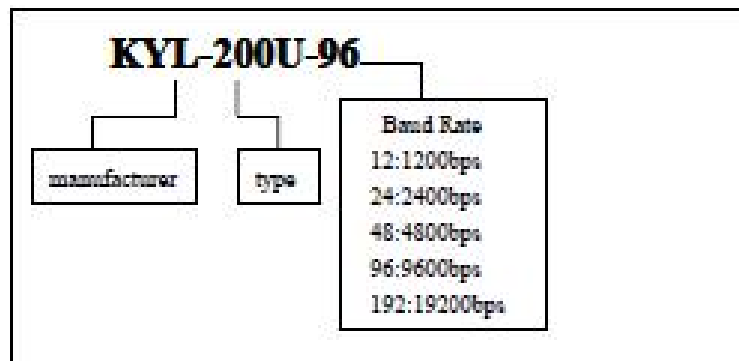
is given to each station. The coordination of communication is controlled by master station that uses data frames containing address code to transmit data or command. Slave station will receive all of the data and command and compare the received address code with local address code. If they are different, the data will be deserted without any response. If those address codes are the same, it means the data is sent to the local. Slave station will make different responses according to the transmitted data or command and send back the data of response. All these jobs must be performed by upper protocol, and it is assured that there is only one transmitter-receiver in the state of transmission in the communication network at any instant moment so as to avoid the cross-interference. KYL-200U can also be used for point-to-point communication with easier operation. For the programming of serial port, all you have to do is to remember that its communication mode is semi duplex while always observing the time sequence of come-and-go for receiving and transmitting.

## **VI. Technical specification of KYL-200U**

- \* Modulation mode: GFSK/FSK
- \* Working frequency: 400MHz~470MHz, 868MHz, 915MHz;
- \* Power supply: DC 5V (3.3~3.6V informed when placing the order);
- \* RF power: ≤10mW, 50mW~100mW is available;
- \* RF data rate: 1200/2400/4800/9600/19200bps (Set before delivery);
- \* Receive current: <20mA (TTL connect);
- \* Receive sensitivity: -120dBm (1200bps); -115dBm (9600bps);
- \* Transmitting current: <40mA;
- \* Sleep current: <20uA;
- \* Interface data format: 8E1/8N1/8O1;
- \* Working humidity: 10%~90% relative humidity without condensation;
- \* Working temperature: -35℃~+75℃ ( industrial)
- \* RF Line-of-sight Range: 600m(BER=10<sup>-5</sup>@9600bps);  
1000m(BER=10<sup>-5</sup>@1200bps);
- \* Size: 47mm\*26mm\*10mm (without antenna port).



## VII. Description of type





**Future Technology Devices International Ltd.**

## **TTL-232R-3V3 USB to TTL Serial Converter Cable**

*The TTL-232R-3V3 is a USB to TTL serial converter cable incorporating FTDI's FT232RL USB - Serial UART interface IC device, the latest device to be added to FTDI's range of USB UART interface Integrated Circuit Devices. It is designed to allow for a fast, simple way to connect devices with a TTL level serial interface to USB.*

*The FT232RL chip used by the TTL-232R-3V3 is housed within the USB 'A' connector. A 1.8 meter (6 foot) cable is terminated with a 6 way 0.1" pitch header socket which provides access to the transmit (Tx), receive (Rx), RTS#, and CTS#, as well as VCC (5V out) and GND.*

*The FT232R is a USB to serial UART interface with optional clock generator output, and the new FTDIChip-ID™ security dongle feature. In addition, asynchronous and synchronous bit bang interface modes are available. USB to serial interface designs using the FT232R have been further simplified by fully integrating the external EEPROM, clock circuit and USB resistors onto the device.*

## 1. Features

### 1.1 Hardware Features

- TTL-232R-3V3 Converter Cable provides a USB to TTL Serial header interface (6 way, 0.1" pitch).
- On board FT232RL provides single chip USB to asynchronous serial data transfer interface.
- Entire USB protocol handled on the FT232RL chip - No USB-specific firmware programming required.
- Connect directly to a microcontroller UART or I/O pins
- UART interface support for 7 or 8 data bits, 1 or 2 stop bits and odd / even / mark / space / no parity.
- Fully assisted hardware (RTS#/CTS#) or X-On / X-Off software handshaking.
- Data transfer rates from 300 baud to 3 Megabaud at TTL levels.
- 3.3V CMOS drive outputs and 5V safe TTL inputs makes the TTL-232R-3V3 easy to interface to 3.3V MCU's.
- FTDI's royalty-free VCP allow for communication as a standard COM port and D2XX 'direct' drivers provide DLL application programming interface.
- In-built support for event characters and line break condition.
- Support for FT232RL FTDIChip-ID™ feature.
- 256 byte receive buffer and 128 byte transmit buffer utilising buffer smoothing technology to allow for high data throughput.
- FIFO receive and transmit buffers for high data throughput.
- Adjustable receive buffer timeout.
- Integrated 1024 bit Internal EEPROM for storing custom USB VID, PID, serial number and product description strings.
- Device supplied preprogrammed with unique USB serial number.
- Support for USB suspend and resume.
- Cable supply voltage comes from USB - no external supply required.
- 5V VCC output allow devices to be powered from the USB port.
- 6 way 0.1" header connector (Tx, Rx, RTS#, CTS#, Vcc, and GND)
- Low USB bandwidth consumption.
- UHCI / OHCI / EHCI host controller compatible
- USB 2.0 Full Speed compatible.
- -40°C to +85°C operating temperature range.
- Connect to a PC via USB 'A' plug on 1.80 meter (6 foot) cable.
- A 5V TTL I/O version is also available (TTL-232R)
- Custom versions also available (subject to MOQ).

### 1.2 Driver Support

#### Royalty-Free VIRTUAL COM PORT (VCP) DRIVERS for...

- Windows 98, 98SE, ME, 2000, Server 2003, XP.
- Windows Vista / Longhorn\*
- Windows XP 64-bit.
- Windows XP Embedded.
- Windows CE.NET 4.2 & 5.0
- MAC OS 8 / 9, OS-X
- Linux 2.4 and greater

#### Royalty-Free D2XX Direct Drivers (USB Drivers + DLL S/W Interface)

- Windows 98, 98SE, ME, 2000, Server 2003, XP.
- Windows Vista / Longhorn\*
- Windows XP 64-bit.
- Windows XP Embedded.
- Windows CE.NET 4.2 & 5.0
- MAC OS OS-X\*
- Linux 2.4 and greater

The drivers listed above are all available to download for free from the [FTDI website](#). Various 3rd Party Drivers are also available for various other operating systems - see the [FTDI website](#) for details.

\* Currently Under Development. Contact FTDI for availability.

### 1.3 Typical Applications

- USB to Serial TTL Level Converter
- Upgrading Legacy Peripherals to USB
- Interface Microcontroller UART or I/O to USB
- Interface FPGA / PLD to USB
- Replace MAX232 type level shifters allowing for direct connection of products to PC via USB
- USB Instrumentation PC Interface
- USB Industrial Control
- USB Software / Hardware Encryption Dongles



## 1. Features

### 1.1 Hardware Features

- TTL-232R-3V3 Converter Cable provides a USB to TTL Serial header interface (5 way, 0.1" pitch).
- On board FT232RL provides single chip USB to asynchronous serial data transfer interface.
- Entire USB protocol handled on the FT232RL chip - No USB-specific firmware programming required.
- Connect directly to a microcontroller UART or I/O pins
- UART interface support for 7 or 8 data bits, 1 or 2 stop bits and odd / even / mark / space / no parity.
- Fully assisted hardware (RTS#/CTS#) or X-On / X-Off software handshaking.
- Data transfer rates from 300 baud to 3 Megabaud at TTL levels.
- 3.3V CMOS drive outputs and 5V safe TTL inputs makes the TTL-232R-3V3 easy to interface to 3.3V MCU's.
- FTDI's royalty-free VCP allow for communication as a standard COM port and D2XX 'direct' drivers provide DLL application programming interface.
- In-built support for event characters and line break condition.
- Support for FT232RL FTDIChip-ID™ feature.
- 256 byte receive buffer and 128 byte transmit buffer utilising buffer smoothing technology to allow for high data throughput.
- FIFO receive and transmit buffers for high data throughput.
- Adjustable receive buffer timeout.
- Integrated 1024 bit internal EEPROM for storing custom USB VID, PID, serial number and product description strings.
- Device supplied preprogrammed with unique USB serial number.
- Support for USB suspend and resume.
- Cable supply voltage comes from USB - no external supply required.
- 5V VCC output allow devices to be powered from the USB port.
- 6 way 0.1" header connector (Tx, Rx, RTS#, CTS#, Vcc, and GND)
- Low USB bandwidth consumption.
- UHCI / OHCI / EHCI host controller compatible
- USB 2.0 Full Speed compatible.
- -40°C to +85°C operating temperature range.
- Connect to a PC via USB 'A' plug on 1.80 meter (6 foot) cable.
- A 5V TTL I/O version is also available (TTL-232R)
- Custom versions also available (subject to MOQ).

### 1.2 Driver Support

#### Royalty-Free VIRTUAL COM PORT (VCP) DRIVERS for...

- Windows 98, 98SE, ME, 2000, Server 2003, XP.
- Windows Vista / Longhorn\*
- Windows XP 64-bit.
- Windows XP Embedded.
- Windows CE.NET 4.2 & 5.0
- MAC OS 8 / 9, OS-X
- Linux 2.4 and greater

#### Royalty-Free D2XX Direct Drivers (USB Drivers + DLL S/W Interface)

- Windows 98, 98SE, ME, 2000, Server 2003, XP.
- Windows Vista / Longhorn\*
- Windows XP 64-bit.
- Windows XP Embedded.
- Windows CE.NET 4.2 & 5.0
- MAC OS OS-X\*
- Linux 2.4 and greater

The drivers listed above are all available to download for free from the [FTDI website](#). Various 3rd Party Drivers are also available for various other operating systems - see the [FTDI website](#) for details.

\* Currently Under Development. Contact FTDI for availability.

### 1.3 Typical Applications

- USB to Serial TTL Level Converter
- Upgrading Legacy Peripherals to USB
- Interface Microcontroller UART or I/O to USB
- Interface FPGA / PLD to USB
- Replace MAX232 type level shifters allowing for direct connection of products to PC via USB
- USB Instrumentation PC Interface
- USB Industrial Control
- USB Software / Hardware Encryption Dongles



Figure 1 - The TTL-232R-3V3 USB to TTL Serial Converter Cable.



Figure 2 - Inside the USB 'A' connector on the TTL-232R-3V3.

### 3. TTL-232R-3V3 Pin Out and Signal Descriptions

#### 3.1 TTL-232R-3V3 Pin Out

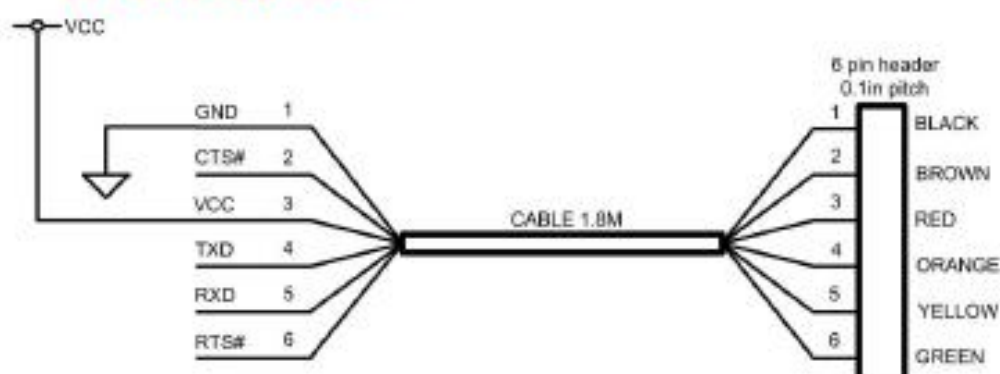


Figure 3 - TTL-232R-3V3 6 Way Header Pin Out

#### 3.2 Serial TTL Signal Descriptions

Table 1 - Serial TTL Signals

Header pin No.	Name	Type	Colour	Description
1	GND	GND	Black	Device ground supply pin.
2	CTS#	Input	Brown	Clear to Send Control Input / Handshake signal.
3	VCC	Output	Red	+5V Output
4	TXD	Output	Orange	Transmit Asynchronous Data output.
5	RXD	Input	Yellow	Receive Asynchronous Data input.
6	RTS#	Output	Green	Request To Send Control Output / Handshake signal.

### 4. TTL-232R-3V3 Parameters

Table 2 - TTL-232R-3V3 I/O Operating Parameters

Parameter	Description	Min	Typ	Max	Units	Conditions
VCC	Output Power Voltage	4.25	5.0	5.25	V	Dependent on the USB port that the TTL-232R-3V3 is connected to
I <sub>o</sub>	Output Power Current	-	-	50	mA	
T	Operating Temperature Range	-40	-	+85	°C	

Table 3 - TTL-232R-3V3 I/O Pin Characteristics

Parameter	Description	Min	Typ	Max	Units	Conditions
V <sub>oh</sub>	Output Voltage High	2.2	2.8	3.2	V	I <sub>source</sub> = 3mA
V <sub>ol</sub>	Output Voltage Low	0.3	0.4	0.6	V	I <sub>sink</sub> = 8mA
V <sub>in</sub>	Input Switching Threshold	1.0	1.2	1.5	V	**
V <sub>Hys</sub>	Input Switching Hysteresis	20	25	30	mV	**

Both the FT232RL I.C. device used and the entire TTL-232R-3V3 are fully compliant with European Union directive 2002/95/EC.

## 5. TTL-232R-3V3 Circuit Schematic

Figure 4 shows the TTL-232R-3V3 circuit schematic. Note that there is an option internal to the cable design which allows for the TTL I/O pins supply voltage to be changed to 5V instead of 3.3V, thus making the TTL-232R-3V3 suitable for interfacing with 5V logic (part number TTL-232R). Customised versions of these cables are also available. Contact [FTDI sales](#) if you are interested in any of these options.

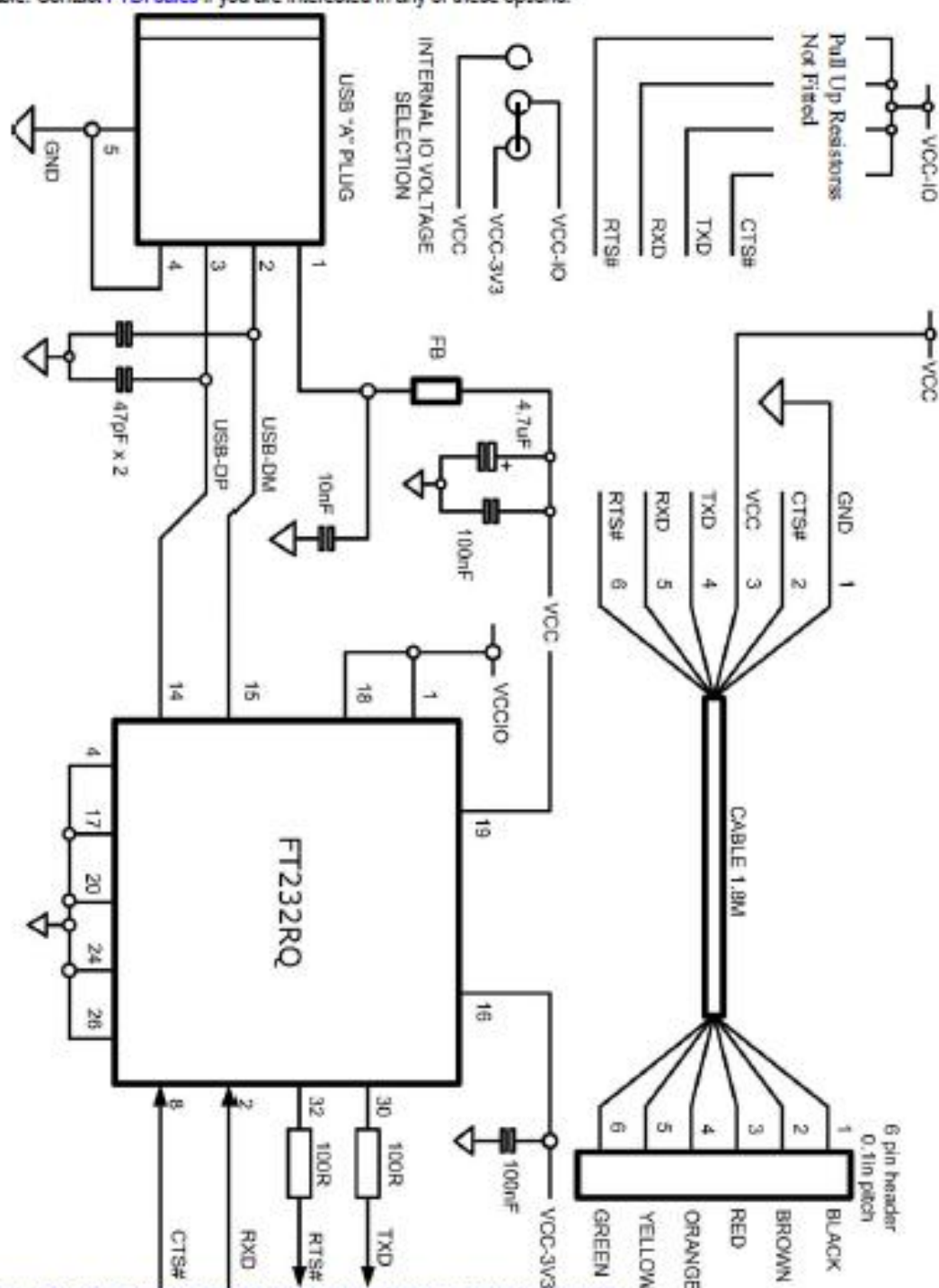


Figure 4 - TTL-232R-3V3 USB to TTL Serial Converter Cable Circuit Schematic

TTL-232R-3V3 USB to TTL Serial Converter Cable (3.3V IO)  
 Datasheet Version 0.90

© Future Technology Devices International Ltd. 2008



## 6. Internal EEPROM Configuration

Following a power-on reset or a USB reset the FT232R will scan its internal EEPROM and read the USB configuration descriptors stored there. The default values programmed into the internal EEPROM in the FT232RQ used in the TTL-232R-3V3 are shown in Table 2.

Table 4 - Default Internal EEPROM Configuration

Parameter	Value	Notes
USB Vendor ID (VID)	0403h	FTDI default VID (hex)
USB Product ID (PID)	8001h	FTDI default PID (hex)
Serial Number Enabled?	Yes	
Serial Number	See Note	A unique serial number is generated and programmed into the EEPROM during final test of the TTL-232R-3V3 module.
Pull Down I/O Pins in USB Suspend	Disabled	Enabling this option will make the device pull down on the UART interface lines when the power is shut off (PWRDN# is High)
Manufacturer Name	FTDI	
Manufacturer ID	FT	
Product Description	TTL232R-3V3	
Max Bus Power Current	90mA	
Power Source	Bus Powered	
Device Type	FT232R	
USB Version	0200	Returns USB 2.0 device descriptor to the host. Note: The device is be a USB 2.0 Full Speed device (12Mbps) as opposed to a USB 2.0 High Speed device (480Mbps).
Remote Wake up	Disabled	RM# pin is not available, so this option has been disabled.
High Current I/Os	Enabled	Enables the high drive level on the UART and CSUS I/O pins
Load VCP Driver	Enabled	Makes the device load the VCP driver interface for the device.
Invert TXD	Disabled	Signal on this pin becomes TXD# if enabled.
Invert RXD	Disabled	Signal on this pin becomes RXD# if enabled.
Invert RTS#	Disabled	Signal on this pin becomes RTS if enabled.
Invert CTS#	Disabled	Signal on this pin becomes CTS if enabled.

The internal EEPROM in the FT232R can be reprogrammed over USB using the utility program MPROG. MPROG can be downloaded from the [FTDI website](#). Version 2.6a or later is required for the FT232R chip. Users who do not have their own USB Vendor ID but who would like to use a unique Product ID in their design can apply to FTDI for a free block of unique PIDs. Contact [FTDI support](#) for this service.



## Disclaimer

---

Copyright © Future Technology Devices International Limited , 2006.

Version 0.9 - Initial Datasheet Created May 2006

Neither the whole nor any part of the information contained in, or the product described in this manual, may be adapted or reproduced in any material or electronic form without the prior written consent of the copyright holder.

This product and its documentation are supplied on an as-is basis and no warranty as to their suitability for any particular purpose is either made or implied.

Future Technology Devices International Ltd. will not accept any claim for damages howsoever arising as a result of use or failure of this product. Your statutory rights are not affected.

This product or any variant of it is not intended for use in any medical appliance, device or system in which the failure of the product might reasonably be expected to result in personal injury.

This document provides preliminary information that may be subject to change without notice.

## Contact FTDI

### Head Office -

Future Technology Devices International Ltd.  
373 Scotland Street,  
Glasgow G5 8QB,  
United Kingdom

Tel. : +(44) 141 429 2777

Fax. : +(44) 141 429 2758

E-Mail (Sales) : [sales1@ftdichip.com](mailto:sales1@ftdichip.com)

E-Mail (Support) : [support1@ftdichip.com](mailto:support1@ftdichip.com)

E-Mail (General Enquiries) : [admin1@ftdichip.com](mailto:admin1@ftdichip.com)

### Regional Sales Offices -

Future Technology Devices International Ltd.  
(Taiwan)  
4F, No 16-1,  
Sec. 6 Minzyuan East Road,  
Neihu District,  
Taipei 114,  
Taiwan, R.O.C.

Tel.: +886 2 8791 3570

Fax: +886 2 8791 3576

E-Mail (Sales): [tw.sales@ftdichip.com](mailto:tw.sales@ftdichip.com)

E-Mail (Support): [tw.support@ftdichip.com](mailto:tw.support@ftdichip.com)

E-Mail (General Enquiries): [tw.admin@ftdichip.com](mailto:tw.admin@ftdichip.com)

Future Technology Devices International Ltd.  
(USA)  
5285 NE Elam Young  
Parkway, Suite B800  
Hillsboro,  
OR 97124-6499  
USA

Tel.: +1 (503) 547-0988

Fax: +1 (503) 547-0987

E-Mail (Sales): [us.sales@ftdichip.com](mailto:us.sales@ftdichip.com)

E-Mail (Support): [us.support@ftdichip.com](mailto:us.support@ftdichip.com)

E-Mail (General Enquiries): [us.admin@ftdichip.com](mailto:us.admin@ftdichip.com)

Website URL : <http://www.ftdichip.com>

**KEPUTUSAN DEKAN  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA  
NOMOR : 175/ELK/Q-I/X/2014  
TENTANG  
PENGANGKATAN PEMBIMBING TUGAS AKHIR D.3  
BAGI MAHASISWA FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

**DEKAN FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA**

Menimbang : 1. Bahwa sehubungan dengan telah dipenuhi syarat untuk penulisan Tugas Akhir D3 bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta, perlu diangkat pembimbing.  
2. Bahwa untuk keperluan dimaksud perlu ditetapkan dengan Keputusan Dekan.

Mengingat : 1. Undang-undang Nomor 20 tahun 2003  
2. Peraturan Pemerintah RI Nomor 60 tahun 1999  
3. Keputusan Presiden RI: a. Nomor 93 tahun 1999; b. 305/M tahun 1999.  
4. Keputusan Menteri Pendidikan dan Kebudayaan RI: Nomor 274/O/1999.  
5. Keputusan Mendiknas RI Nomor 003/O/2001.  
6. Keputusan Rektor UNY Nomor : 1160/UN34/KP/2011

**MEMUTUSKAN**

**Menetapkan**

Pertama : Mengangkat Pembimbing Tugas Akhir D3 bagi mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta sebagai berikut :

Nama Pembimbing : Dessy Irmawati, MT  
Bagi mahasiswa  
Nama/No. Mahasiswa : Enggar Prajangga / 10507131027  
Jurusan / Prodi : Pendidikan Teknik Elektronika / Teknik Elektronika  
Judul Tugas Akhir : Prototipe Telemetry Alat Monitoring Cuaca Menggunakan AT mega 8 dengan Antarmuka Komputer

Kedua : Dosen pembimbing disertai tugas membimbing penulisan Tugas Akhir sesuai dengan Pedoman Tugas Akhir.

Ketiga : Keputusan ini berlaku sejak ditetapkan

Keempat : Segala sesuatu akan diubah dan dibetulkan sebagaimana mestinya apabila di kemudian hari ternyata terdapat kekeliruan dalam Keputusan ini.

Ditetapkan : di Yogyakarta  
Pada tanggal : 1 Oktober 2014



Dekan  
Dr. Moch. Bruri Triyono  
NIP. 19560216 198603 1 003

**Tembusan Yth :**

1. Wakil Dekan II FT UNY
2. Ketua Jurusan Pendidikan Teknik Elektronika
3. Kasub. Bag. Pendidikan FT UNY
4. Yang bersangkutan.



JURUSAN PENDIDIKAN TEKNIK ELEKTRONIKA  
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA  
Alamat : Kampus Karangmalang Yogyakarta 55281  
Telp. : (0274) 554686 ; (0274) 586168 ext. 293



KARTU BIMBINGAN PROYEK AKHIR (Untuk Mahasiswa)

FRM/EKA/03-00  
25 Januari 2008

Nama Mahasiswa :  
No. Mahasiswa :  
Program Studi \*) :  
Kelas :  
Dosen Pembimbing :  
Judul :

ENIGGAR PRANANGGA  
10507131027  
PFTeknik Elektronika  
B2  
DESSY IMAWATI, MT No. Telp. / HP : 081328015688  
PROTOTYPE TELEMETRI ALAT PENDETEKSI DINI  
KEBAKARAN HUTAN MENGGUNAKAN ATMEGA 8  
DENGAN ANTARMUKA KOMPUTER

Lejang : D3/84

No	Tanggal	Urutan Bimbingan	Tandatangan Pembimbing
1.	15/3/2014	Revisi Bab I	Chafri
2.	12/1/2015	Revisi Bab I	Chafri
3.	21/1/2015	Revisi, lanjut bab II	Chafri
4.	26/1/2015	Revisi Bab I & II	Chafri
5.	11/3/2015	Revisi Bab I & II	Chafri
6.	27/3/15	Revisi, lanjut Panduan	Chafri
7.	15/4/15	Revisi, lanjut Panduan & penulisan	Chafri
8.	28/5/15	Revisi yg dikasir mesin cetak	Chafri
9.	22/6/15	Revisi	Chafri
10.	24/6/15	Revisi, final pengisian	Chafri

Rekomendasi Pembimbing :

1. Mahasiswa yang bersangkutan siap untuk diuji.
2. Kartu Bimbingan ini wajib dilampirkan pada saat pendataan ujian Proyek Akhir.

Tanggal Persetujuan :

Tandatangan Pembimbing :



JURUSAN PENDIDIKAN TEKNIK ELEKTRONIKA  
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA  
Alamat : Kampus Karangmalang Yogyakarta 55281  
Telp. : (0274) 554686 ; 586168 ext. 293



KARTU MONITORING PROYEK AKHIR (Untuk Dosen)

FRM/EKA/04-00  
25 Januari 2008

Nama Mahasiswa :  
No. Mahasiswa :  
Program Studi \*) :  
Kelas :  
Dosen Pembimbing :  
Judul :

ENIGGAR PRANANGGA  
10507131027 No. Telp. / HP : 08132511293  
PFTeknik Elektronika  
B2  
DESSY IMAWATI, MT  
PROTOTYPE TELEMETRI ALAT PENDETEKSI DINI  
KEBAKARAN HUTAN MENGGUNAKAN ATMEGA 8  
DENGAN ANTARMUKA KOMPUTER

No	Tanggal	Catatan Pembimbing
1.		
2.		
3.		
4.		
5.		
6.		
7.		
8.		
9.		
10.		

Keterangan :

Mahasiswa yang bersangkutan telah disetujui untuk ujian Proyek Akhir.

Tanggal Persetujuan :

Tandatangan :



JURUSAN PENDIDIKAN TEKNIK ELEKTRONIKA  
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA  
Alamat : Kampus Karangmalang Yogyakarta 55281  
Telp. : (0274) 554686 : (0274) 586168 ext. 293



KARTU BIMBINGAN PROYEK AKHIR (Untuk Mahasiswa)  
FRM/EKA/03-00  
25 Januari 2008

Nama Mahasiswa : ENGGA PRAMANGA  
No. Mahasiswa : 10507131027  
Program Studi \*) : PT-Elektronika/Teknik Elektronika  
Kelas : 8.2  
Dosen Pembimbing : DESSY RIMAWATI, S.T.Ng. Telp. / HP : 08132801888  
Judul : PROTOTYPE ALAT PENDETEKSI DINI  
KEBAKARAN HUTAN MENGGUNAKAN ATMEGA8  
DENCAN ANTARMUKA KOMPUTER

No	Tanggal	Uraian Bimbingan	Tandatangan Pembimbing
1.	18/8/15	Revisi	Chy
2.	07/10/15	Acc	Chy
3.			
4.			
5.			
6.			
7.			
8.			
9.			
10.			

**Rekomendasi Pembimbing :**

1. Mahasiswa yang bersangkutan siap untuk diuji.
2. Kartu Bimbingan ini wajib ditempelkan pada saat pendaharan ujian Proyek Akhir.

Tanggal Persetujuan : 7/10/15 Tandatangan Pembimbing : Chy



JURUSAN PENDIDIKAN TEKNIK ELEKTRONIKA  
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA  
Alamat : Kampus Karangmalang Yogyakarta 55281  
Telp. : (0274) 554686 : 586168 ext. 293



KARTU MONITORING PROYEK AKHIR (Untuk Dosen)  
FRM/EKA/04-00  
25 Januari 2008

Nama Mahasiswa : ENGGA PRAMANGA  
No. Mahasiswa : 10507131027  
Program Studi \*) : PT-Elektronika/Teknik Elektronika  
Kelas : 8.2  
Dosen Pembimbing : DESSY RIMAWATI, S.T., M.T.  
Judul : PROTOTYPE ALAT PENDETEKSI DINI  
KEBAKARAN HUTAN MENGGUNAKAN ATMEGA8  
DENCAN ANTARMUKA KOMPUTER

No	Tanggal	Catatan Pembimbing
1.		
2.		
3.		
4.		
5.		
6.		
7.		
8.		
9.		
10.		

**Keterangan :**

Mahasiswa yang bersangkutan telah disetujui untuk ujian Proyek Akhir.

Tanggal Persetujuan : 7/10/15 Tandatangan : Chy





KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN  
UNIVERSITAS NEGERI YOGYAKARTA

**FAKULTAS TEKNIK**

ALAMAT : Karangmalang Yogyakarta 55281  
Telepon (0274) 586734 (Dekan), (0274) 586168 psw, 276, 292, 289

**KETENTUAN WAKTU UJIAN TUGAS AKHIR SKRIPSI  
DAN SUSUNAN PENGUJI**

Berdasarkan Surat Keterangan Bebas Teori Kasubag. Pendidikan Fakultas Teknik Universitas Negeri Yogyakarta Nomor 578m/EKA/TAP3/2015 Tanggal 02 Juli 2015

Nama : ENGGAR PRADANEGA  
No. Mahasiswa : 10507131027  
Jurusan : TEKNIK ELEKTRONIKA D3  
Angkatan tahun : 2010  
Judul yang diajukan : PROTOTYPE TELEMETRI ALAT PENDETEKSI DINI  
KEBAKARAN HUTAN MENGGUNAKAN ATMEGA 8  
DENGAN ANTARMUKA KOMPUTER

Dosen Pembimbing : 1. DESSY IRMAWATI, S.T., M.T.  
Ujian akan dilaksanakan pada :  
Hari, tanggal : Jum'at, 10 Juli 2015  
Jam : 13.00  
Tempat : Ruang Sidang FT Universitas Negeri Yogyakarta

Susunan Penguji sebagai berikut :

1. Ketua : DESSY IRMAWATI, S.T., M.T.  
2. Sekretaris : SATRIYO AGUNG DEWANTO, M.Pd.  
3. Penguji : Dr. EKO MARRANAJI

Demikian agar dapat diproses untuk selanjutnya.

Yogyakarta, 03 Juli 2015

Ketua Jurusan

(Dr. MUHAMMAD MUNIR, M.Pd.)  
NIP.19630512 198901 1 001

Catatan  
Dibuat rangkap dua

1. Serahkan Subag. Akademik
2. Arsip Jurusan

**KEPUTUSAN DEKAN  
FAKULTAS TEKNIK  
UNIVERSITAS NEGERI YOGYAKARTA  
NOMOR : 22/EKA-TA/XI/2015  
TENTANG  
PENGANGKATAN PANITIA PENGUJI TUGAS AKHIR D-3  
BAGI  
MAHASISWA FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA  
ATAS NAMA : Enggar Prajangga**

**DEKAN FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA**

- Menimbang : 1. Bahwa sehubungan dengan telah dipenuhinya persyaratan untuk mengikuti ujian Tugas Akhir D-3 bagi mahasiswa F.T. UNIVERSITAS NEGERI YOGYAKARTA, dipandang perlu untuk dilaksanakan ujian Tugas Akhir D-3 dengan tertib dan lancar serta penentuan hasilnya dapat dinilai secara obyektif.
2. Bahwa untuk keperluan dimaksud dipandang perlu mengangkat Panitia Penguji Tugas Akhir D-3 dengan Keputusan Dekan.
- Mengingat : 1. Undang-Undang RI : Nomor 20 Tahun 2003
2. Peraturan Pemerintah RI : Nomor 60 Tahun 1999
3. Keputusan Presiden RI : Nomor 93 Tahun 1999 ; Nomor 305 M Tahun 1999
4. Keputusan Menteri Pendidikan dan Kebudayaan RI Nomor 0464/O/1992 ; Nomor 274/O/1999
5. Keputusan Menteri Pendidikan Nasional RI : Nomor 003/0/2001
6. Keputusan Rektor UNY : Nomor 1160/UN34/KP/2011
- Mengingat pula : Keputusan Dekan Fakultas Teknik UNIVERSITAS NEGERI YOGYAKARTA Nomor 042 Tahun 1989
- MEMUTUSKAN**
- Menetapkan  
Pertama : Mengangkat Panitia Penguji Tugas Akhir D-3 bagi mahasiswa Fakultas Teknik UNIVERSITAS NEGERI YOGYAKARTA yang susunan personalianya sebagai berikut :
1. Ketua : Dessy Irmawati, MT
2. Sekretaris : Satriyo Agung Dewanto, M.Pd
3. Penguji Utama : Dr. Eko Marpanaji
- Bagi mahasiswa :
- Nama/No. Mahasiswa : **Enggar Prajangga /10507131027**
- Jurusan/Prodi : Pendidikan Teknik Elektronika / Teknik Elektronika
- Judul Tugas Akhir : *Prototipe Telemetry Alat Pendeteksi Dini Kebakaran Hutan Menggunakan AT mega 8 dengan Antarmuka Komputer*
- Kedua : Ujian dilaksanakan pada hari Jum'at, tanggal 10 Juli 2015 mulai pukul 13.00 sampai dengan selesai, bertempat di ruang Sidang.
- Ketiga : Segala sesuatu akan diubah dan dibetulkan sebagaimana mestinya apabila dikemudian hari ternyata terdapat kekeliruan dalam Keputusan ini.

Ditetapkan : di Yogyakarta  
Pada tanggal : 3 Juli 2015

Dekan  
Dr. Moch. Bruri Triyono  
NIP. 19560216 198603 1 003

- Tembusan Yth :
1. Wakil Dekan II FT UNY
  2. Ketua Jurusan Pendidikan Teknik Elektronika
  3. Kasub. Bag. Pendidikan FT UNY
  4. Kepala Media FT UNY
  5. Yang bersangkutan.